KD11-K

BASIC LOGIC TEST

MD-11-DQKDA-B

EP-DQKDA-B-DL-A

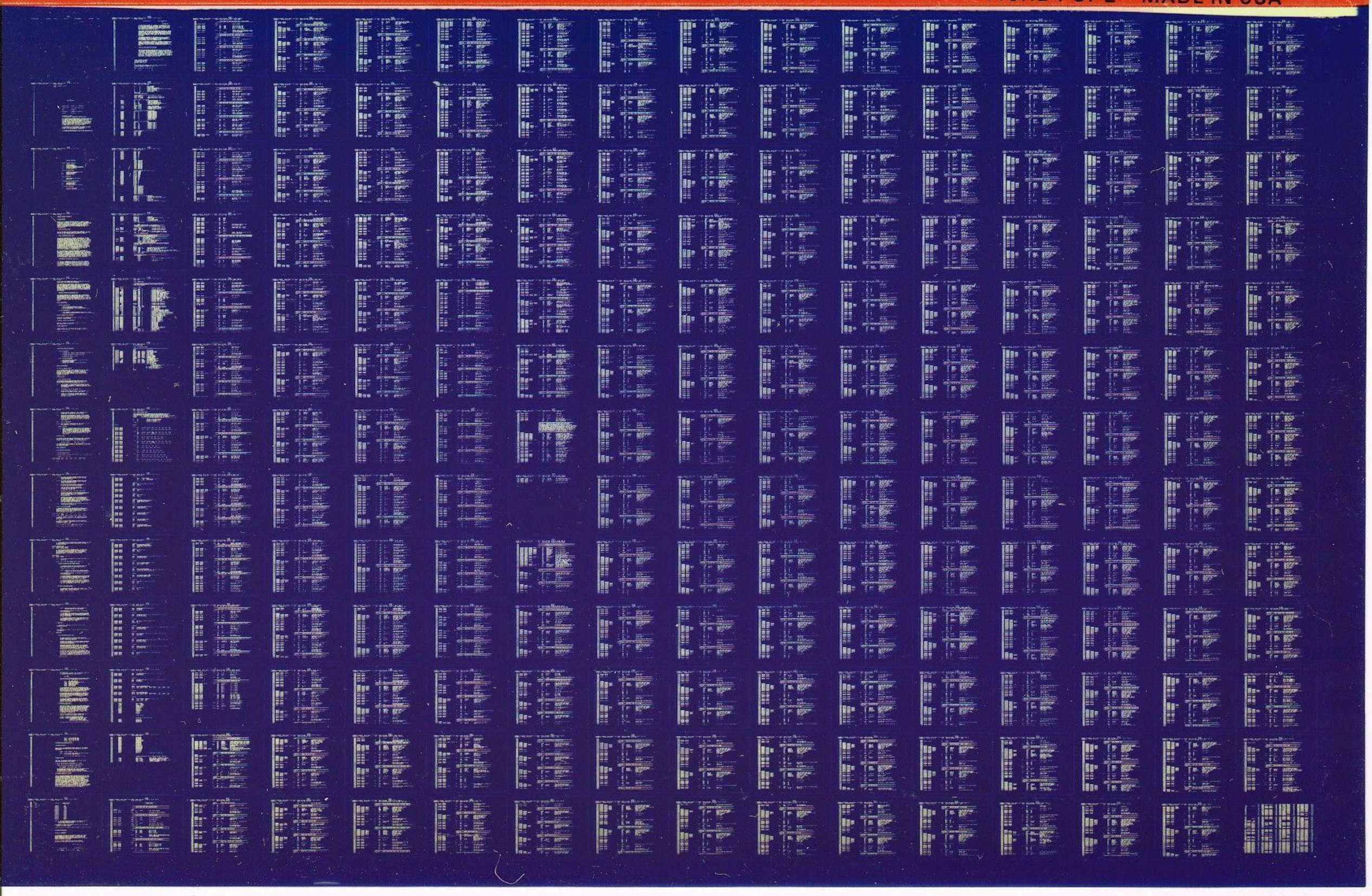
COPYRIGHT © 1977

FICHE 1 OF 2

JUN 1977

digital

MADE IN USA



KD11-K

BASIC LOGIC TEST

MD-11-DQKDA-B

EP-DQKDA-B-DL-A

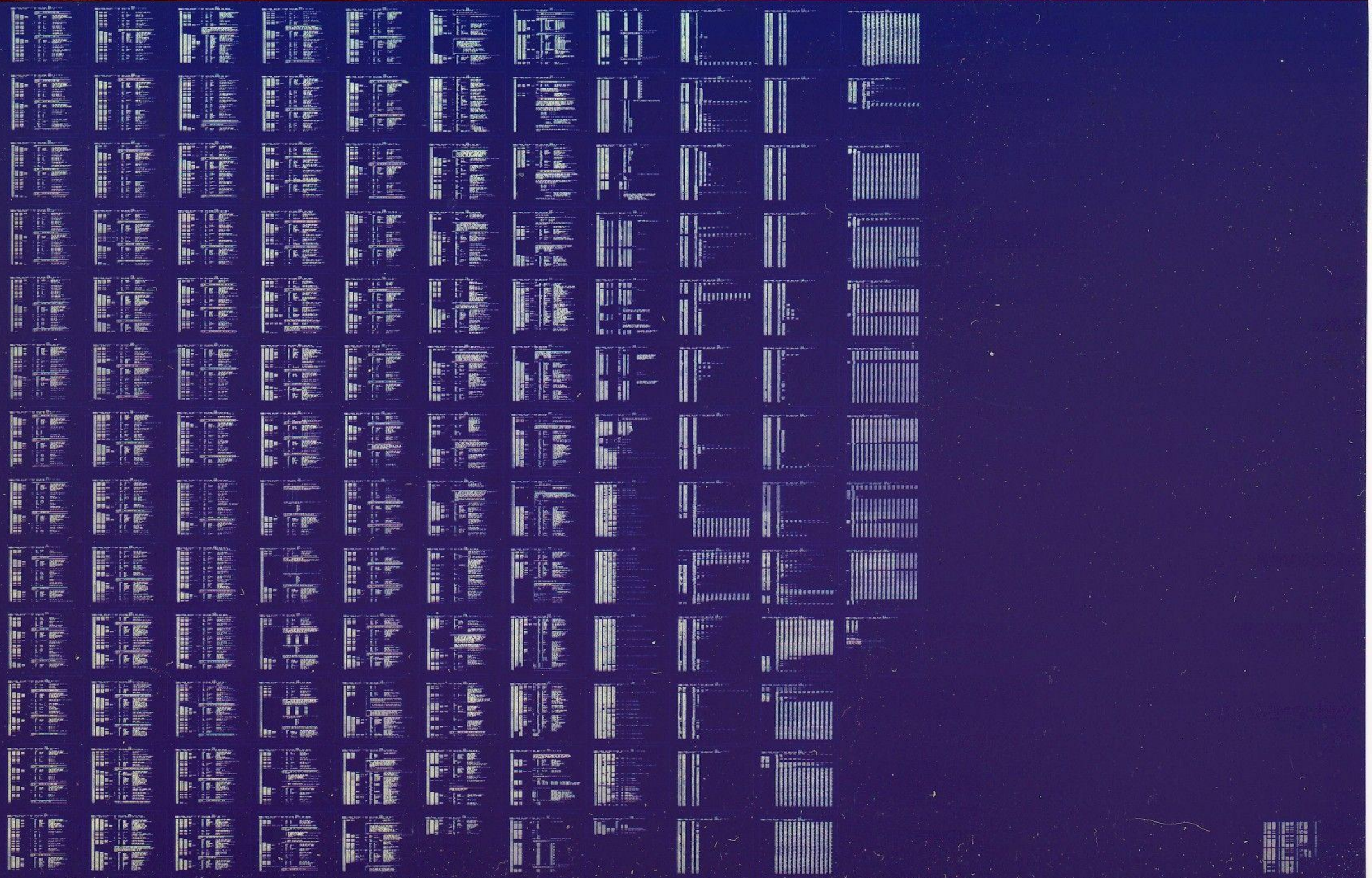
COPYRIGHT © 1977

FICHE 2 OF 2

JUN 1977

digital

MADE IN USA



B01

E0F1D2LPKGSE0	00010000	770624	PDP10 411	HDR1D0KD0B5E0	00010000	770624

```
                                .SBTTL  DOCUMENT LISTING
                                .REM %
```

IDENTIFICATION

PRODUCT CODE: MAINDEC - 11 - DQKDA-B-D

PRODUCT NAME: KD11-K BASIC LOGIC TESTS

DATE: 30-MAR-77

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: JACK RICH

COPYRIGHT (C) 1977
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY OTHER COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO AGREES TO TO THESE LICENSE TERMS. TITLE TO AND OWNERSHIP OF THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.

TABLE OF CONTENTS

1.0 GENERAL PROGRAM INFORMATION

1.1 PROGRAM PURPOSE
1.2 SYSTEM REQUIREMENTS
1.3 RELATED DOCUMENTS AND STANDARDS
1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
1.5 FAILURE ASSUMPTIONS

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES
2.2 SPECIAL ENVIRONMENTS
2.3 PROGRAM OPTIONS
2.4 EXECUTION TIMES

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES
3.2 ERROR HALTS

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS
4.2 PROGRESS REPORTS
4.3 MAINTENANCE BREAKPOINT FEATURE

5.0 MAINTENANCE PROCEDURES

5.1 THE KD11-K PROCESSOR
5.2 CONDITION CODE SCOPE SYNC FEATURE

1.0 GENERAL PROGRAM INFORMATION

1.1 PROGRAM PURPOSE

"DQKDA" IS A DIAGNOSTIC PROGRAM DESIGNED TO DETECT, REPORT, AND IDENTIFY LOGIC FAULTS IN THE KD11-K CENTRAL PROCESSING UNIT OF THE PDP11/6X SYSTEM. IT CONSISTS OF 504(10) INDIVIDUAL TESTS CAREFULLY DESIGNED AND SEQUENCED TO DETECT AND ATTEMPT TO IDENTIFY LOGIC FAULTS AT A MINIMUM HARDWARE/SOFTWARE LEVEL. THESE TESTS ARE PARTITIONED INTO FOUR MAJOR SECTIONS AS DESCRIBED BELOW:

A. BASIC CPU TESTS (BCPT)

THIS IS THE BASIC CPU TEST TO VERIFY THE "HARDCORE". ANY FAULT DETECTED IN THIS SECTION CAUSES THE PROGRAM TO HALT WITH THE PC+2 OF THE HALT INSTRUCTION DISPLAYED ON THE CONSOLE.

B. BASIC INSTRUCTION TESTS (BIT)

THIS SECTION CONSISTS OF A LOGICALLY SEQUENCED SET OF BASIC INSTRUCTION TESTS DESIGNED TO VERIFY THE INTEGRITY OF THOSE INSTRUCTIONS AND LOGIC OPERATIONS USED BY THE UTILITY ROUTINES THAT PROVIDE ERROR LOGGING AND SCOPE LOOPING FACILITIES FOR THE SUBSEQUENT TWO MAJOR SECTIONS. NO UTILITY IS CALLED UNTIL ITS INSTRUCTION COMPLEMENT HAS BEEN VERIFIED. THIS SCHEME ACCOMPLISHES TWO IMPORTANT MAINTENANCE OBJECTIVES: 1)IT MINIMIZES THE POSSIBILITY OF THE ERROR REPORTING ROUTINES CONVEYING AMBIGUOUS ERROR INFORMATION TO THE USER, AND 2)IT MAXIMIZES THE POSSIBILITY THAT THE ERROR WILL BE DETECTED BY A ROUTINE DESIGNED TO IDENTIFY FAILING OPERATIONS RATHER THAN HAVE THE ERROR MANIFEST ITSELF IN A MORE COMPLEX UTILITY ROUTINE THAT IS NOT STRUCTURED TO DIAGNOSE FAULTS.

ANY FAULT DETECTED IN THIS SECTION CAUSES THE PROGRAM TO HALT WITH THE CONSOLE ADDRESS INDICATING THE PC+2 OF THE HALT INSTRUCTION IN THE FAILING TEST. ADDITIONAL FAULT IDENTIFICATION INFORMATION IS AVAILABLE IN THE PROCESSOR'S GENERAL REGISTERS, PSW, STACK, AND PROGRAM ANNOTATION FOR THE FAILING TEST. A LOCK ON HARD ERROR FEATURE IS EMPLOYED TO PREVENT THE PROGRAM FROM CONTINUING ON ONCE A SOLID ERROR IS DETECTED. DEPRESSING CONTINUE AFTER THE ERROR HALT CAUSES A RETRY OF THE FAILING TEST.

C. COMPREHENSIVE INSTRUCTION TESTS (CIT)

THIS SECTION, COMPRISED OF THE BULK OF THE TESTS, CONSISTS OF A LOGICALLY SEQUENCED AND PARTITIONED SET OF INSTRUCTION TESTS DESIGNED TO TEST AND VERIFY ALL THE BASIC INSTRUCTIONS OF THE KD11-K PROCESSOR. THIS EXCLUDES TESTING THOSE LOGIC FUNCTIONS THAT SUPPORT THE CONSOLE FUNCTIONS (LOAD ADDRESS, DEPOSIT, ETC.). EACH TEST IN THIS SECTION CALLS A "SCOPE LOOP"

UTILITY THAT FACILITATES USER CONTROL OF TEST SELECTION AND EXECUTION VIA THE CONSOLE SWITCH REGISTER.

UPON DETECTION OF A LOGIC FAULT, EACH TEST IN THIS SECTION CALLS AN "ERROR SERVICE" ROUTINE THAT LOGS THE ERROR AND REPORTS IT AS HARD COPY ON THE CONSOLE TERMINAL DEVICE. THE ERROR SERVICE ROUTINE ALSO FACILITATES USER CONTROL OF THE PROGRAM SEQUENCE VIA CONSOLE SWITCH REGISTER OPTIONS. AFTER REPORTING THE ERROR THE PROGRAM CONTINUES ON IN ITS NORMAL SEQUENCE UNLESS MODIFIED BY THE USER ACTIVATING THE "LOCK ON HARD ERROR" SWITCH OPTION.

D. COMBINED INSTRUCTION EXERCISER (IEX)

THIS SECTION CONSISTS OF A MORE COMPLEX SET OF INSTRUCTION TESTS DESIGNED TO TEST THE INSTRUCTIONS WHEN USED IN VARIOUS COMBINATIONS MANIPULATING VARIABLE DATA PATTERNS. IT ALSO TESTS THE MED AND ERROR LOGGING FEATURES OF THE CPU. LIKE THE PREVIOUS SECTION, IT CALLS THE "ERROR SERVICE" AND "SCOPE LOOP" UTILITIES TO REPORT ERRORS AND ALLOW USER CONTROL OF TEST EXECUTION.

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

1. PDP11/6X CPU WITH OPERATOR'S CONSOLE
2. 16K OF CORE STORAGE - MF11/U OR EQUIVALENT
3. DL11-W ASYNCHRONOUS LINE INTERFACE WITH LINE CLOCK

B. SOFTWARE REQUIREMENTS

1. PDP11 ABSOLUTE LOADER PROGRAM FOR PAPER TAPE SYSTEMS
2. XXDP MONITOR FOR DECTAPE,MAGTAPE,CASSETTE, OR DISK SYSTEMS.

1.3 RELATED DOCUMENTS AND STANDARDS

DQKDA USES THE STANDARD APT SOFTWARE INTERFACES FOUND IN THE MACY11 SYSMAC PACKAGES.

1.4 DIAGNOSTIC HIERARCHY REQUIREMENTS

DQKDA WILL NORMALLY BE THE FIRST DIAGNOSTIC TO BE RUN AS PART OF PDP 11/6X CPU CHECKOUT.

1.5 FAILURE ASSUMPTIONS

"DQKDA" ASSUMES THAT THE STORAGE MEDIUM USED TO STORE THE PROGRAM IS INTACT AND THAT IT CAN BE LOADED INTO CORE.

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURES

A. LOADING PROCEDURES

1) STANDARD PDP11 ABSOLUTE LOADER PROCEDURES FOR PAPER TAPE.

2) STANDARD XXDP MONITOR LOADING PROCEDURES.

3) STANDARD APT OR ACT LOADING

B. MANUAL STARTING PROCEDURES

1) LOAD SWITCH REG WITH 000000 (NO SWITCH OPTIONS)
2) SET DISPLAY TO 000200
3) DEPRESS LOAD ADDRESS
4) PRESS CNTRL AND START BUTTONS SIMULTANEOUSLY

2.2 SPECIAL ENVIRONMENTS

16K PDP11/6X SERIES SYSTEMS

FOR 16K SYSTEMS USING THE "XXDP" PACKAGE YOU WILL BE UNABLE TO USE THE "UPDATE" PROGRAMS TO LOAD,SAVE, UPDATE ETC. SINCE THE SIZE OF "DQKDA" WILL NOT PERMIT SIMULTANEOUS RESIDENCY OF THE UPDATE PROGRAMS. SUFFICIENT FREE CORE IS AVAILABLE FOR THE "XXDP" MONITOR SO THAT "DQKDA" CAN BE LOADED BY THE MONITOR.

2.3 PROGRAM OPTIONS

A. SWITCH REGISTER OPTIONS

THE FOLLOWING CONSOLE SWITCH REGISTER OPTIONS ARE ACTIVE UPON ENTERING THE COMPREHENSIVE INSTRUCTION TESTS (CIT) SECTION: (SWITCH OPTION IS ACTIVE WHEN SW IS SET TO A "1")

SW15 HALT ON ERROR. IF ERROR PRINTING IS ENABLED THE HALT OCCURS AFTER THE PRINTOUT. DEPRESSING "CONTINUE" CAUSES THE PROGRAM TO PROCEED ON IN NORMAL SEQUENCE FROM THE POINT OF ERROR.

SW14 CONTINUOUSLY LOOP ON THE CURRENT TEST

SW13 INHIBIT NORMAL ERROR PRINTOUTS - THIS DOES NOT INCLUDE POWER FAIL, BUS ERROR, OR RSVD INSTR TRAPS.

SW12 INHIBIT ALL PRINTOUTS NOT COVERED UNDER SW13. THIS INCLUDES I.D., BUS ERROR, AND RSVD INSTR TRAPS. NOTE THAT IT IS NOT POSSIBLE TO INHIBIT END PASS OR POWER FAIL PRINTOUTS.

SW11 INHIBIT SUB-TEST ITERATIONS. TEST ITERATIONS ARE AUTOMATICALLY INHIBITED ON THE FIRST PASS.

SW10 SEARCH FOR AND CONTINUOUSLY LOOP ON THE TEST NUMBER SELECTED BY THE CONTENTS OF SW<08:00>. ONLY USE THIS OPTION FOR TESTS TST176 THRU TST767 SINCE THE "SCOPE" UTILITY IS NOT ACTIVE UNTIL TEST TST176. LOOPING ON TST176 WILL CAUSE A LOOP ON THE ENTIRE "BIT" SECTION (TESTS 0-176).

SW09 LOCK ON HARD ERROR

SW<8:0> USED TO SELECT A PARTICULAR TEST FOR LOOPING IF SW10=1. TEST NUMBER MUST BE BETWEEN 176 AND 767.

B. MEMORY LOCATIONS

4. BPTLOC: THERE IS A LOCATION TAGGED "BPTLOC" THAT PROVIDES THE USER THE MECHANISM FOR SETTING SIXTEEN "BREAKPOINT HALTS" THROUGHOUT THE PROGRAM. THIS ENABLES RAPIDLY "HOMING IN" ON THE FAILING TEST IN THOSE CASES WHERE THE FAULT CAUSES A RUNAWAY OR HUNG PROGRAM. REFER TO PARA. 4.2 FOR A DETAILED DESCRIPTION OF THE USE OF THIS FEATURE.

2.4 EXECUTION TIMES

ONE COMPLETE ERROR FREE PASS OF DQKDA WITH NO TEST ITERATIONS SHOULD TAKE LESS THAN 7 SECONDS. A SUCCESSFUL PASS WILL BE INDICATED BY THE FOLLOWING PRINTOUT ON THE CONSOLE DEVICE:

END PASS # 000001 ERROR COUNT = 000000

THIS ERROR COUNT IS NOT CLEARED AT THE BEGINNING OF A NEW PASS. WITH ITERATIONS ENABLED A COMPLETE ERROR FREE PASS SHOULD TAKE LESS THAN 2.5 MINUTES.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES

A. ERROR MESSAGE FORMATS

THERE ARE SEVERAL DIFFERENT ERROR FORMATS. EACH IS DESCRIBED BELOW.

1.) ERROR 1 IS OF THE FORM

S/B DST	WAS DST	DEST	(IR)	TEST	(PC)	(SP)	(PSW)
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX

WHERE:

```
                        S/B DST FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS WHAT THE RESULT (DEST. OPERAND)
                                SHOULD HAVE BEEN (S/B).

                        WAS DST FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS WHAT THE RESULT (DEST. OPERAND)
                                ACTUALLY WAS AFTER THE TEST.

                        DEST    FOR SINGLE AND DOUBLE OPERAND INSTRUCTIONS THIS
                                COLUMN CONTAINS THE DESTINATION ADDRESS.

                        (IR)    THIS IS A COPY OF THE TEST INSTRUCTION.
                                THIS WILL BE THE FIRST WORD IN THE CASE OF TWO
                                OR THREE WORD INSTRUCTIONS.

                        TEST    INDICATES THE TEST NO. (IN OCTAL) THAT FAILED

                        (PC)    INDICATES THE CONTENTS OF THE PROGRAM COUNTER AT THE
                                TIME OF THE ERROR CALL. THIS IS AN ADDRESS NORMALLY
                                USED TO LOCATE THE ERROR CALL STATEMENT IN
                                THE FAILING TEST.

                        (SP)    INDICATES THE CONTENTS OF THE STACK POINTER (R6) AT
                                THE TIME OF THE ERROR. NOTE THAT THE ERROR CALL
                                WILL PUSH THE STACK TWICE.  IN SP TESTS WHERE THE
                                SP MUST BE RESTORED PRIOR TO CALLING THE ERROR ROUTINE,
                                THEN THE ORIGINAL (UNRESTORED) SP IS TYPED, WITHOUT
                                ADDITIONAL PUSHES FROM THE ERROR CALL.

                        (PSW)   INDICATES THE CONTENTS OF THE PROCESSOR STATUS WORD
                                AT THE TIME OF THE ERROR CALL

                        XXXXXX  IS AN OCTAL NUMBER.

                        2.) ERROR 2 AND ERROR 4 ARE THE SAME AS FOR ERROR 1 ABOVE
                        EXCEPT THAT IN THIS CASE THE DESTINATION IS A GENERAL
                        REGISTER (WHICH DOES NOT HAVE A UNIBUS ADDRESS).  THE OCTAL
                        NUMBER TYPED OUT IN THE "DEST" COLUMN SHOULD BE IGNORED.
                        THE TYPOUT WOULD LOOK AS FOLLOWS:

S/B DST WAS DST  DEST    (IR)    TEST    (PC)    (SP)    (PSW)
                 IS R3
XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX

                        3.) ERROR 5, ERROR 6, AND ERROR 7 ARE IDENTICAL TO ERROR 1
                        EXCEPT THAT ONLY THE LAST 5,6, OR 7 COLUMNS (RESPECTIVELY)
                        ARE PRINTED.

                        4.) ERROR 3 IS USED IN CASES WHERE THE STACK POINTER IS
                        SPECIFICALLY IN ERROR.  THE COLUMNS HAVE THE SAME MEANING AS
                        DESCRIBED FOR ERROR 1 EXCEPT:

                        S/B SP  IS WHAT THE STACK POINTER SHOULD HAVE BEEN (S/B)

                        WAS SP  IS WHAT THE STACK POINTER ACTUALLY WAS
```

```
                        5.) OTHER ERRORS TYPE OUT THEIR SPECIFIC ERROR MESSAGE, FOLLOWED
                        BY SELF EXPLANATORY DATA HEADERS, DE  NDING ON THE ERROR.
                        AN EXAMPLE FOLLOWS:

BAD DATA READ BY A MED
  PC     MEDCODE EXPECTD RECEIVD
XXXXXX  XXXXXX  XXXXXX  XXXXXX

                        6.) WHEN THE SCOPE ROUTINE BECOMES ACTIVE, IT CHECKS THAT
                        THE TEST NUMBER (IN R0) IS EXACTLY ONE GREATER THAN THE
                        TEST NUMBER ON THE PREVIOUS SCOPE CALL.  IF A MACHINE
                        ERROR CAUSES TESTS TO BE SKIPPED, OR THE PROGRAM TO JUMP
                        BACKWARDS, ERROR 11 WILL REPORT THIS AS FOLLOWS:

TESTS SKIPPED
  PC     EXPCTD  ACTUAL  (TEST #'S)
XXXXXX  XXXXXX  XXXXXX

                        EXPCTD  THIS IS THE TEST NUMBER THE SCOPE WAS EXPECTING TO BE
                                CALLED FROM.

                        ACTUAL  THIS IS THE TEST NUMBER THAT IT FOUND IN R0

                        7.) RESERVED INSTRUCTION TRAP ERROR MESSAGE
                            ---------------------------------------

                            ANY RESERVED INSTRUCTION TRAP DETECTED AFTER THE
                            BASIC TESTS RESULTS IN THE FOLLOWING PRINTOUT:

                            TRAPPED TO 10  PC = XXXXXX

                            WHERE:  XXXXXX IS THE VALUE OF THE PROGRAM COUNTER
                                    PUSHED ON THE STACK WHEN THE TRAP WAS SPRUNG.

                            AFTER REPORTING THE ERROR, THE PROGRAM IS RESTARTED
                            FROM THE BEGINNING.

                            IF A RSVD INSTRUCTION TRAP OCCURS WHILE IN THE PROCESS
                            OF TRYING TO SERVICE A PREVIOUS RSVD INSTRUCTION TRAP
                            OR A BUS ERROR TRAP THE PROGRAM HALTS. A DESCRIPTION
                            OF THIS HALT IS CONTAINED IN PARA. 3.2.3 BELOW.

                            IF A RSVD INSTRUCTION TRAP OCCURS PRIOR TO COMPLETION
                            OF THE BASIC INSTRUCTION TEST SECTION THE PROGRAM WILL
                            HALT VIA A TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF
                            THIS HALT IS DESCRIBED IN PARA. 3.2.2 BELOW.


                    4. BUS ERROR TRAP ERROR MESSAGE
                       ----------------------------

                            ANY UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD
                            ADDRESS ERROR, ILLEGAL INSTRUCTION, OR STACK OVERFLOW)
                            RESULTS IN THE FOLLOWING PRINTOUT:

                            TRAPPED TO 4  PC = XXXXXX
```

```
                                        WHERE:  XXXXXX IS THE VALUE OF THE PC PUSHED ONTO
                                                THE STACK WHEN THE TRAP WAS SPRUNG.

                                        AFTER REPORTING THE ERROR THE PROGRAM IS RESTARTED
                                        FROM THE BEGINNING.

                                        IF A BUS ERROR TRAP OCCURS WHILE A PREVIOUS BUS ERROR
                                        OR RSVD INSTRUCTION IS STILL PENDING THE PROGRAM WILL
                                        HALT. A DESCRIPTION OF THE HALT INTERPRETATION IS GIVEN
                                        IN PARA. 3.2.3 BELOW.

                                        IF A BUS _RROR OCCURS PRIOR TO THE COMPLETION OF THE
                                        BASIC INSTRUCTION TESTS, THE PROGRAM WILL HALT VIA A
                                        TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF THIS HALT
                                        IS INCLUDED IN PARA. 3.2.2 BELOW.

                                    5. POWER FAIL
                                       ------------

                                       IF A POWER FAIL CONDITION IS DETECTED, THE FOLLOWING
                                       MESSAGE IS PRINTED:

                                            POWER

                                       AFTER PRINTING AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
                                       THE BEGINNING.

                            3.2     ERROR HALTS
                                    -----------

                                    1. BASIC INSTRUCTION TESTS (BIT)
                                       -----------------------------

                                            ANY ERROR DETECTED IN THE BASIC TESTS CAUSES THE
                                            PROGRAM TO HALT WITH THE PC+2 OF THE LOCATION CONTAINING
                                            THE HALT INSTRUCTION DISPLAYED.

                                            EXAMINING THE CONTENTS OF THE CPU'S GENERAL REGISTERS,
                                            THE PSW, AND THE STACK WILL PROVIDE ADDITIONAL FAULT
                                            IDENTIFICATION INFORMATION.

                                            DEPRESSING "CONTINUE" AFTER THE HALT WILL CAUSE AN
                                            AUTOMATIC RETRY OF THE FAILING TEST. IF THE ERROR IS
                                            SOLID THE PROGRAM WILL LOCK ON THIS TEST, BUT IF IT
                                            IS INTERMITTENT THE PROGRAM WILL CONTINUE ON IN NORMAL
                                            SEQUENCE ONCE THE TEST IS SUCCESSFULLY EXECUTED.

                                            TO ESTABLISH A TIGHT SCOPE LOOP ON THE FAILING TEST,
                                            REPLACE THE "HALT" WITH A 400(8). AND DEPRESS "CONTINUE"
                                            THE "400" IS A "BR .+2" WHICH FUNCTIONS AS A NOP. THIS
                                            IS NECESSARY TO PRESERVE THE INTEGRITY OF THE CONDITION
                                            CODE OPERATE INSTRUCTION THAT IS USED AS A SCOPE SYNC. THIS
                                            BUILT IN SYNC FEATURE IS DESCRIBED IN PARA. 5.0.

                                    2. TRAPCATCHER HALTS
```

```
                                        ------------------
                                          THE VECTOR AREA (LOC 000 - 776) IS PROGRAM LOADED WITH
                                          A STANDARD TRAPCATCHER AS SHOWN BELOW:

                                                  V  / V+2
                                                  V+2/ HALT

                                          AFTER THE BASIC INSTRUCTION TESTS THE FOLLOWING VECTORS
                                          ARE SET UP TO POINT TO APPROPRIATE SERVICE ROUTINES:

                                                  4/6     BUS ERROR SERVICE
                                                  10/12   RSVD INSTRUCTION TRAP SERVICE
                                                  20/22   SCOPE LOOP SERVICE
                                                  24/26   POWER FAIL SERVICE
                                                  30/32   ERROR SERVICE
                                                  34/36   PRINT SERVICE

                                          AT THE APPROPRIATE POINTS IN THE COMPREHENSIVE INSTR-
                                          UCTION TESTS THE LINE CLOCK VECTOR (100/102) AND THE DL11
                                          VECTORS (60/62 - 64/66) ARE SET UP TO CHECK INTERRUPTS
                                          FROM THESE DEVICES. ALL OTHER VECTORS REMAIN SET UP TO
                                          "CATCH" UNEXPECTED TRAPS OR INTERRUPTS BY HALTING.

                                          WHEN AN UNEXPECTED TRAP OR INTERRUPT NOT SUPPORTED BY
                                          AN APPROPRIATE SERVICE ROUTINE OCCURS THE CPU HALTS.
                                          WITH THE PC+4 OF THE VECTOR DISPLAYED IN THE CONSOLE.
                                          THIS IS USED TO IDENTIFY THE CAUSE OF THE UNEXPECTED
                                          TRAP OR INTERRUPT.

                                          THE LAST ENTRY PUSHED ON THE STACK CAN BE EXAMINED
                                          TO DETERMINE WHERE THE PROGRAM WAS WHEN THE TRAP OR
                                          INTERRUPT WAS SPRUNG. REMEMBER THAT THE "OLD PC" GETS
                                          SAVED ON THE STACK WHEN A TRAP OR INTERRUPT OCCURS.

                                  3. CATASTROPHIC ERROR HALTS
                                     ------------------------

                                          THERE ARE TWO HALTS, ONE IN THE BUS ERROR SERVICE ROU-
                                          TINE AND THE OTHER IN THE RSVD INSTRUCTION TRAP SERVICE
                                          ROUTINE THAT HALT THE PROGRAM IF ONE OF THESE ERRORS
                                          OCCURS WHILE STILL SERVICING A PREVIOUS BUS ERROR
                                          OR RSVD INSTRUCTION TRAP. AFTER THE HALT THE CONSOLE
                                          DISPLAYS THE PC+2 OF THE ERROR HALT. THIS IS USED
                                          TO IDENTIFY WHICH OF THE TWO TYPES OF ERRORS - RSVD
                                          OR BUS ERROR - OCCURRED LAST.

                                          THERE IS A SOFTWARE FLAG TAGGED "CATERR" THAT MAY BE
                                          EXAMINED TO OBTAIN THE FOLLOWING INFORMATION:

                                          [CATERR] = 000002 TWO SUCCESSIVE BUS ERRORS
                                          [CATERR] = 001000 TWO SUCCESSIVE RSVD INSTR. TRAPS
                                          [CATERR] = 000401 A COMBINATION OF THE TWO. THE
                                                            CONTENTS OF THE ADDRESS DISPLAY
                                                            IDENTIFIES WHICH TYPE OCCURRED LAST.

                                          THE STACK PROVIDES THE FOLLOWING ADDITIONAL INFORMATION:
```

```
                                        [SP  ] / PC OF THE 2ND TRAP
                                        [SP+2] / PSW OF THE 2ND TRAP
                                        [SP+4] / PC OF THE 1ST TRAP
                                        [SP+6] / PSW OF THE 1ST TRAP

                        4.0     PERFORMANCE AND PROGRESS REPORTS
                                --------------------------------

                        4.1     PERFORMANCE REPORTS
                                -------------------

                                THERE IS ONLY ONE PERFORMANCE REPORT SUPPLIED BY THE PROGRAM
                                AND CONSISTS OF A SIMPLE END OF PASS MESSAGE OF THE FORMAT
                                SHOWN BELOW:

                                        PASCNT = XXXXXX  ERRCNT = YYYYYY

                                        WHERE:  XXXXXX IS THE TOTAL NUMBER OF COMPLETE PASSES
                                                OF THE ENTIRE PROGRAM (OCTAL)

                                                YYYYYY IS THE TOTAL ERROR COUNT IN OCTAL

                        4.2     PROGRESS REPORTS
                                ----------------

                                THERE ARE TWO PROGRESS REPORTS PRINTED THAT REPORT NORMAL
                                ERROR FREE EXECUTION OF THE PROGRAM.

                                A. END OF PASS PRINTOUT AS DESCRIBED IN 4.1 ABOVE.

                                B. PROGRAM IDENTIFICATION MESSAGE AS DESCRIBED BELOW:

                                        MD-11-D0KDAB   KD11-K BASIC LOGIC TESTS

                                   THIS MESSAGE GETS PRINTED THE FIRST TIME THE PROGRAM ENTERS
                                   THE COMPREHENSIVE INSTRUCTION TEST SECTION UNLESS
                                   INHIBITED BY SW12=1. AFTER THE FIRST PASS THIS PRINTOUT
                                   IS AUTOMATICALLY INHIBITED UNLESS THE PROGRAM IS RESTARTED AT 200(8).

                        4.3     MAINTENANCE BREAKPOINT FEATURE
                                ------------------------------

                                THERE IS A MANUAL PROGRESS REPORT FEATURE THAT ALLOWS THE USER
                                TO STEP THROUGH THE PROGRAM, HALTING AFTER EVERY N'TH TEST WITH
                                PROGRESS INFORMATION DISPLAYED IN THE CONSOLE ADDRESS DISPLAYS.
                                TO ACTIVATE THIS FEATURE THE USER MUST SET THE DESIRED
                                "BREAKPOINT HALT" BITS IN THE MEMORY LOCATION TAGGED "BPTLOC".
                                THIS LOCATION PROVIDES SIXTEEN POSSIBLE HALTS DISPERSED EVENLY
                                THROUGHOUT THE PROGRAM (APPROX. EVERY 20 TESTS). AT EACH CHECK-
                                POINT THE PROGRAM EXAMINES A PARTICULAR BIT IN "BPTLOC" AND
                                HALTS IF THE BIT IS SET TO A "1" OTHERWISE IT CONTINUES IN
                                NORMAL SEQUENCE. AFTER THE HALT DEPRESSING "CONTINUE" WILL CAUSE
                                RESUMPTION OF NORMAL PROGRAM EXECUTION. SETTING LOCATION "BPTLOC"
                                TO ALL 1'S (177777) WILL RESULT IN THE FOLLOWING SIXTEEN HALTS
                                WITH THE INFORMATION SHOWN DISPLAYED IN THE CONSOLE:
```

[BPTLOC]	ADDRESS DISPLAY HALT PC+2
BIT00=1	4326
BIT01=1	6312
BIT02=1	10632
BIT03=1	11762
BIT04=1	14356
BIT05=1	17116
BIT06=1	21542
BIT07=1	24350
BIT08=1	27162
BIT09=1	32156
BIT10=1	34642
BIT11=1	37452
BIT12=1	42142
BIT13=1	46142
BIT14=1	52602
BIT15=1	55426

NOTE: IF THE USER DEPOSITED A 000400(8) IN LOCATION "BPTLOC"
ONLY ONE HALT WOULD OCCUR AND AT THAT TIME THE
DISPLAY SHOULD CONTAIN 27162.

THIS FEATURE IS USEFUL FOR TRACKING DOWN THE TEST THAT CAUSES
A "RUNAWAY" OR "HUNG" PROGRAM.

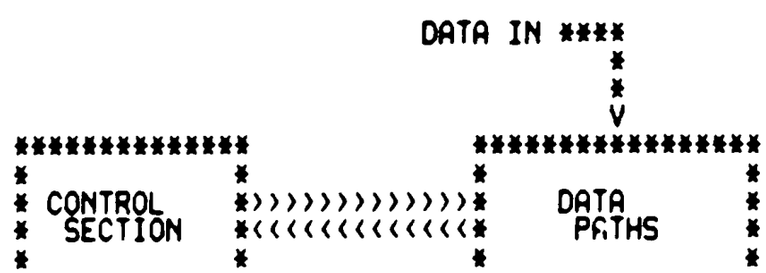
LOCATION "BPTLOC" IS PROGRAM LOADED AS 000000 TO INHIBIT ANY HALTS.

5.0 MAINTENANCE PROCEDURES

5.1 THE KD11-K PROCESSOR

THE PROCEDURES OUTLINED IN THIS SECTION ASSUME THAT "DQKDA"
CAN BE LOADED INTO CORE AND STARTED. IF THE FAILURE MODE
PREVENTS PROGRAM LOADING OR AFFECTS NORMAL POWER UP AND
CONSOLE OPERATIONS, THE TECHNICIAN MUST REVERT TO THE MANUAL
DEBUG AND CHECKOUT PROCEDURES.

THE KD11-K CENTRAL PROCESSING UNIT CAN BE VIEWED AS
CONSISTING OF TWO MAJOR LOGIC AREAS AS DEPICTED BELOW:



```
660                                          **************          ****************
661                                                                          *
662                                                                          *
663                                                                          ***>DATA OUT
664
665                                     THE DATA PATHS CONSIST OF A LOGICALLY INTERCONNECTED GROUP
666                                     OF STATIC DATA FACILITIES (REGISTERS, MULTIPLEXORS, ALU'S
667                                     ETC.) REQUIRED TO TEMPORARILY STORE, MODIFY, AND TRANSFER
668                                     DATA ITEMS (16 BIT WORDS OR 8 BIT BYTES) ACCORDING TO THE
669                                     DESIGN SPECIFICATIONS FOR THE PDP11.
670
671                                     THE CONTROL SECTION SUPPLIES PREDEFINED SEQUENCES OF CONTROL
672                                     SIGNAL SETS TO ACTIVATE THE REQUIRED DATA FACILITIES WITHIN
673                                     THE DATA PATHS. IN THE KD11-K THESE CONTROL SIGNAL SETS ARE
674                                     STORED IN A READ ONLY MEMORY (ROM) AND GENERATED BY READING
675                                     OUT A UNIQUE SEQUENCE OF ROM WORDS FOR EACH OPERATION TO BE
676                                     PERFORMED.
677
678                                     THE SEQUENCE GENERATED BY THE CONTROL SECTION IS VARIABLE
679                                     AND DEPENDENT UPON THE INSTRUCTION OR LOGIC OPERATION
680                                     BEING EXECUTED. THERE ARE HUNDREDS OF THESE SEQUENCES POSS-
681                                     IBLE DEPENDENT UPON OF THE PROGRAM CODING.
682
683                                     "D0KDA" IS DESIGNED TO GENERATE ALL POSSIBLE MICROINSTRUCTION
684                                     SEQUENCES AND COMBINATIONS OF DATA AND CONTROL SIGNALS. THE
685                                     INDIVIDUAL TESTS ARE LOGICALLY SEQUENCED AND STRUCTURED TO
686                                     DETECT AND ISOLATE PARTICULAR MICROPROGRAM SEQUENCES THAT ARE
687                                     FAULTY.
688
689                                5.2  CONDITION CODE SCOPE SYNC FEATURE
690                                     ---------------------------------
691
692                                     FROM THE BIT SECTION TO THE MED TESTS IN THE CIT SECTION, ALL
693                                     TEST INSTRUCTIONS ARE PRECEEDED BY A CONDITION CODE OPERATE
694                                     INSTRUCTION.  THE UBREAK REGISTER IS PROGRAM LOADED TO GENERATE
695                                     A SYNC PULSE NEAR THE END OF THIS INSTRUCTION.  DURING THE MED
696                                     TESTS, THE PULSE IS GENERATED NEAR THE BEGINNING OF THE MED
697                                     EXECUTION.  THIS PULSE IS GENERATED ON BACKPLANE PIN B03M2 AND
698                                     MAY BE USED IN CONJUNTION WITH THE PROGRAM LOOPING FEATURES TO
699                                     PROBE THE KD11-K DURING THE FAILING TEST.
700
701                                %
702
703                                .TITLE  MAINDEC-11-D0KDA-B  KD11-K BASIC LOGIC TESTS
704                                ;*COPYRIGHT (C) JAN 1977
705                                ;*DIGITAL EQUIPMENT CORP.
706                                ;*MAYNARD, MASS. 01754
707                                ;*
708                                ;*
709                                ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
710                                ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
711                                ;*
712                                .SBTTL  OPERATIONAL SWITCH SETTINGS
713                                ;*
714                                ;*      SWITCH                  USE
715                                ;*      ------          --------------------
```

```
                                ;*          15          HALT ON ERROR
                                ;*          14          LOOP ON TEST
                                ;*          13          INHIBIT ERROR TYPEOUTS
                                ;*          12          INHIBIT ID MESSAGE & UNEXPECTED TRAP MESSAGES
                                ;*          11          INHIBIT ITERATIONS
                                ;*          10          LOOP ON TEST IN SWR<8:0>
                                ;*           9          LOOP ON ERROR
                                .ENABLE ABS
                                .SBTTL  BASIC DEFINITIONS

                                ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
001000                          STACK=  1000
                                .EQUIV  EMT,ERROR       ;;BASIC DEFINITION OF ERROR CALL
                                .EQUIV  IOT,SCOPE       ;;BASIC DEFINITION OF SCOPE CALL

                                ;*MISCELLANEOUS DEFINITIONS
000011                          HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                          LF=     12              ;;CODE FOR LINE FEED
000015                          CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                          CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                          PS=     177776          ;;PROCESSOR STATUS WORD
                                .EQUIV  PS,PSW
177774                          STKLMT= 177774          ;;STACK LIMIT REGISTER
177772                          PIRQ=   177772          ;;PROGRAM INTERRUPT REQUEST REGISTER
177570                          DSWR=   177570          ;;HARDWARE SWITCH REGISTER
177570                          DDISP=  177570          ;;HARDWARE DISPLAY REGISTER

                                ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                          R0=     %0              ;;GENERAL REGISTER
000001                          R1=     %1              ;;GENERAL REGISTER
000002                          R2=     %2              ;;GENERAL REGISTER
000003                          R3=     %3              ;;GENERAL REGISTER
000004                          R4=     %4              ;;GENERAL REGISTER
000005                          R5=     %5              ;;GENERAL REGISTER
000006                          R6=     %6              ;;GENERAL REGISTER
000007                          R7=     %7              ;;GENERAL REGISTER
000006                          SP=     %6              ;;STACK POINTER
000007                          PC=     %7              ;;PROGRAM COUNTER

                                ;*PRIORITY LEVEL DEFINITIONS
000000                          PR0=    0               ;;PRIORITY LEVEL 0
000040                          PR1=    40              ;;PRIORITY LEVEL 1
000100                          PR2=    100             ;;PRIORITY LEVEL 2
000140                          PR3=    140             ;;PRIORITY LEVEL 3
000200                          PR4=    200             ;;PRIORITY LEVEL 4
000240                          PR5=    240             ;;PRIORITY LEVEL 5
000300                          PR6=    300             ;;PRIORITY LEVEL 6
000340                          PR7=    340             ;;PRIORITY LEVEL 7

                                ;*"SWITCH REGISTER" SWITCH DEFINITIONS
100000                          SW15=   100000
040000                          SW14=   40000
020000                          SW13=   20000
010000                          SW12=   10000
004000                          SW11=   4000
002000                          SW10=   2000
```

```
                001000          SW09=   1000
                000400          SW08=   400
                000200          SW07=   200
                000100          SW06=   100
                000040          SW05=   40
                000020          SW04=   20
                000010          SW03=   10
                000004          SW02=   4
                000002          SW01=   2
                000001          SW00=   1
                                .EQUIV  SW09,SW9
                                .EQUIV  SW08,SW8
                                .EQUIV  SW07,SW7
                                .EQUIV  SW06,SW6
                                .EQUIV  SW05,SW5
                                .EQUIV  SW04,SW4
                                .EQUIV  SW03,SW3
                                .EQUIV  SW02,SW2
                                .EQUIV  SW01,SW1
                                .EQUIV  SW00,SW0

                                ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
                100000          BIT15=  100000
                040000          BIT14=  40000
                020000          BIT13=  20000
                010000          BIT12=  10000
                004000          BIT11=  4000
                002000          BIT10=  2000
                001000          BIT09=  1000
                000400          BIT08=  400
                000200          BIT07=  200
                000100          BIT06=  100
                000040          BIT05=  40
                000020          BIT04=  20
                000010          BIT03=  10
                000004          BIT02=  4
                000002          BIT01=  2
                000001          BIT00=  1
                                .EQUIV  BIT09,BIT9
                                .EQUIV  BIT08,BIT8
                                .EQUIV  BIT07,BIT7
                                .EQUIV  BIT06,BIT6
                                .EQUIV  BIT05,BIT5
                                .EQUIV  BIT04,BIT4
                                .EQUIV  BIT03,BIT3
                                .EQUIV  BIT02,BIT2
                                .EQUIV  BIT01,BIT1
                                .EQUIV  BIT00,BIT0

                                ;*BASIC "CPU" TRAP VECTOR ADDRESSES
                000004          ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
                000010          RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
                000014          TBITVEC=14              ;;"T" BIT
                000014          TRTVEC= 14              ;;TRACE TRAP
                000014          BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
                000020          IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
```

```
                        000024                  PWRVEC= 24              ;;POWER FAIL
                        000030                  EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
                        000034                  TRAPVEC=34              ;;"TRAP" TRAP
                        000060                  TKVEC=  60              ;;TTY KEYBOARD VECTOR
                        000064                  TPVEC=  64              ;;TTY PRINTER VECTOR
                        000240                  PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
                                                .SBTTL  TRAP CATCHER

                        000000                          .=0
                                                ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
                                                ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
                                                ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
                        000174                          .=174
000174  000000                                  DISPREG: .WORD  0               ;;SOFTWARE DISPLAY REGISTER
000176  000000                                  SWREG:   .WORD  0               ;;SOFTWARE SWITCH REGISTER
                                                .SBTTL  STARTING ADDRESS(ES)
000200  000137  001630                                  JMP     @#START ;;JUMP TO STARTING ADDRESS OF PROGRAM
                        000700                  .=700                           ;PUT APT HEADER IN STACK AREA
                                                .SBTTL  APT PARAMETER BLOCK

                                                ;;**************************************************************
                                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
                                                ;;**************************************************************
                        000700                          .$X=.   ;;SAVE CURRENT LOCATION
                        000024                          .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                                          200     ;;FOR APT START UP
                        000044                          .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  000700                                          $APTHDR ;;POINT TO APT HEADER BLOCK
                        000700                          .=.$X   ;;RESET LOCATION COUNTER
                                                ;;**************************************************************
                                                ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                                ;INTERFACE SPEC.

000700                                          $APTHD:
000700  000000                                  $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
000702  001120                                  $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
000704  000000                                  $TSTM:  .WORD           ;;RUN TIM OF LONGEST TEST
000706  000000                                  $PASTM: .WORD           ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000710  000000                                  $UNITM: .WORD           ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000712  000014                                          .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
                                                .SBTTL  ACT11 HOOKS

                                                ;;**************************************************************
                                                ;HOOKS REQUIRED BY ACT11
                        000714                          $SVPC=.                 ;SAVE PC
                        000046                          .=46
000046  060644                                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
                        000052                          .=52
000052  000000                                          .WORD   0               ;;2)SET LOC.52 TO ZERO
                        000714                          .=$SVPC                 ;; RESTORE PC
```

```
                                .SBTTL  COMMON TAGS

                                ;;**********************************************************************
                                ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
                                ;*USED IN THE PROGRAM.

                        001000          .=1000
001000                  $CMTAG:                         ;;START OF COMMON TAGS
001000  000000                  .WORD   0
001002  000             $TSTNM: .BYTE   0               ;;CONTAINS THE TEST NUMBER
001003  000             $ERFLG: .BYTE   0               ;;CONTAINS ERROR FLAG
001004  000000          $ICNT:  .WORD   0               ;;CONTAINS SUBTEST ITERATION COUNT
001006  000000          $LPADR: .WORD   0               ;;CONTAINS SCOPE LOOP ADDRESS
001010  000000          $LPERR: .WORD   0               ;;CONTAINS SCOPE RETURN FOR ERRORS
001012  000000          $ERTTL: .WORD   0               ;;CONTAINS TOTAL ERRORS DETECTED
001014  000             $ITEMB: .BYTE   0               ;;CONTAINS ITEM CONTROL BYTE
001015  001             $ERMAX: .BYTE   1               ;;CONTAINS MAX. ERRORS PER TEST
001016  000000          $ERRPC: .WORD   0               ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001020  000000          $GDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'GOOD' DATA
001022  000000          $BDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'BAD' DATA
001024  000000          $GDDAT: .WORD   0               ;;CONTAINS 'GOOD' DATA
001026  000000          $BDDAT: .WORD   0               ;;CONTAINS 'BAD' DATA
001030  000000                  .WORD   0               ;;RESERVED--NOT TO BE USED
001032  000000                  .WORD   0
001034  000             $AUTOB: .BYTE   0               ;;AUTOMATIC MODE INDICATOR
001035  000             $INTAG: .BYTE   0               ;;INTERRUPT MODE INDICATOR
001036  000000                  .WORD   0
001040  177570          SWR:    .WORD   DSWR            ;;ADDRESS OF SWITCH REGISTER
001042  177570          DISPLAY: .WORD  DDISP           ;;ADDRESS OF DISPLAY REGISTER
                        $TKS:   177560                  ;;TTY KBD STATUS
                        $TKB:   177562                  ;;TTY KBD BUFFER
                        $TPS:   177564                  ;;TTY PRINTER STATUS REG. ADDRESS
                        $TPB:   177566                  ;;TTY PRINTER BUFFER REG. ADDRESS
001054  000             $NULL:  .BYTE   0               ;;CONTAINS NULL CHARACTER FOR FILLS
001055  002             $FILLS: .BYTE   2               ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001056  012             $FILLC: .BYTE   12              ;;INSERT FILL CHARS. AFTER A "LINE FEED"
001057  000             $TPFLG: .BYTE   0               ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
001060  000000          $REGAD: .WORD   0               ;;CONTAINS THE ADDRESS FROM
                                                        ;;WHICH  ($REG0) WAS OBTAINED
001062  000000          $REG0:  .WORD   0               ;;CONTAINS (($REGAD)+0)
001064  000000          $REG1:  .WORD   0               ;;CONTAINS (($REGAD)+2)
001066  000000          $REG2:  .WORD   0               ;;CONTAINS (($REGAD)+4)
001070  000000          $REG3:  .WORD   0               ;;CONTAINS (($REGAD)+6)
001072  000000          $REG4:  .WORD   0               ;;CONTAINS (($REGAD)+10)
001074  000000          $REG5:  .WORD   0               ;;CONTAINS (($REGAD)+12)
001076  000000          $TMP0:  .WORD   0               ;;USER DEFINED
001100  000000          $TMP1:  .WORD   0               ;;USER DEFINED
001102  000000          $TMP2:  .WORD   0               ;;USER DEFINED
001104  000000          $TMP3:  .WORD   0               ;;USER DEFINED
001106  000000          $TMP4:  .WORD   0               ;;USER DEFINED
001110  000000          $TIMES: 0                       ;;MAX. NUMBER OF ITERATIONS
001112  000000          $ESCAPE:0                       ;;ESCAPE ON ERROR ADDRESS
001114  077             $QUES:  .ASCII  /?/             ;;QUESTION MARK
001115  015             $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
001116  000012          $LF:    .ASCIZ  <12>            ;;LINE FEED
                        ;;**********************************************************************
```

```
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;*****************************************************************
                                .EVEN
001120                          $MAIL:                          ;;APT MAILBOX
001120  000000                  $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
001122  000000                  $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
001124  000000                  $TESTN: .WORD   ATESTN          ;;TEST NUMBER
001126  000000                  $PASS:  .WORD   APASS           ;;PASS COUNT
001130  000000                  $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
001132  000000                  $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
001134  000000                  $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
001136  000000                  $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
001140                          $ETABLE:                        ;;APT ENVIRONMENT TABLE
001140     000                  $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
001141     000                  $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
001142  000000                  $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
001144  000000                  $USWR:  .WORD   AUSWR           ;;USER SWITCHES
001146  000000                  $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
                                ;*                              BITS 15-11=CPU TYPE
                                ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
                                ;*                                      11/70=06,PDQ=07,Q=10
                                ;*                              BIT 10=REAL TIME CLOCK
                                ;*                              BIT  9=FLOATING POINT PROCESSOR
                                ;*                              BIT  8=MEMORY MANAGEMENT
001150                          $ETEND:
                                .MEXIT
```

```
                                .SBTTL  ERROR POINTER TABLE

                                ;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
                                ;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
                                ;*LOCATION $ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
                                ;*NOTE1:        IF $ITEMB IS 0 THE ONLY PERTINENT DATA IS ($ERRPC).
                                ;*NOTE2:        EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

                                ;*      EM              ;;POINTS TO THE ERROR MESSAGE
                                ;*      DH              ;;POINTS TO THE DATA HEADER
                                ;*      DT              ;;POINTS TO THE DATA
                                ;*      DF              ;;POINTS TO THE DATA FORMAT


001150                          $ERRTB:

                                ;ITEM 1
001150  064640                          EM1     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001152  000000                          0
001154  067774                          DT1     ;$REG4,  $REG3,   $REG2, $REG1,$REG0,$ERRPC,$REG5,$REG6
001156  000000                          0
                                ;ITEM 2
001160  064640                          EM2     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001162  065061                          DH2     ;                    IS R3
001164  067774                          DT2     ;$REG4,  $REG3,  $REG2, $REG1, $REG0,$ERRPC,$REG5,$REG6
001166  000000                          0
                                ;ITEM 3
001170  065013                          EM3     ;S/B SP  WAS SP  (IR)  TEST  (PC)  (PSW)
001172  000000                          0
001174  070016                          DT3     ;$REG4,  $REG3,  $REG1,$REG0,$ERRPC,$REG6
001176  000000                          0
                                ;ITEM 4
001200  064640                          EM4     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001202  065072                          DH4     ;                    IS R5
001204  067774                          DT4     ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001206  000000                          0
                                ;ITEM 5
001210  064666                          EM5     ;(IR)  TEST  (PC)  (SP)  (PSW)
001212  000000                          0
001214  070002                          DT5     ;$REG1, $REG0, $ERRPC, $REG5, $REG6
001216  000000                          0
                                ;ITEM 6
001220  064660                          EM6     ;  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001222  000000                          0
001224  070000                          DT6     ;$REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001226  000000                          0
                                ;ITEM 7
001230  064650                          EM7     ;WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001232  000000                          0
001234  067776                          DT7     ;$REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001236  000000                          0
                                ;ITEM 10
001240  064724                          EM10    ;S/B RES  WAS RES  DST OP  STC OP  TEST  (PC) (SP) (PSW)
001242  000000                          0
001244  067774                          DT10    ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001246  000000                          0
```

```
1017                                    ;ITEM 11
1018  001250  065271                        EM11    ;TESTS SKIPPED
1019  001252  065307                        DH11    ;  PC    EXPCTD ACTUAL (TEST #'S)
1020  001254  070034                        DT11    ;$ERRPC, $TESTN,$REG0
1021  001256  000000                        0
1022
1023                                    ;ITEM 12
1024  001260  065345                    EM12    ;MED DID NOT ABORT IN USER MODE
1025  001262  067403                    DH23    ;PC
1026  001264  067674                    DT23    ;$ERRPC
1027  001266  000000                    0
1028
1029                                    ;ITEM 13
1030  001270  065404                    EM13    ;MED EXECUTED IN USER MODE
1031  001272  067403                    DH23    ;PC
1032  001274  067674                    DT23    ;$ERRPC
1033  001276  000000                    0
1034
1035                                    ;ITEM 14
1036  001300  065436                    EM14    ;MED CHANGED PSW
1037  001302  067403                    DH23    ;PC
1038  001304  067674                    DT23    ;$ERRPC
1039  001306  000000                    0
1040
1041                                    ;ITEM 15
1042  001310  065456                    EM15    ;MICROBREAK TRAP-TO-4 DID NOT OCCUR
1043  001312  067314                    DH15    ;ERRPC MEDCODE MICROBK REG.
1044  001314  067640                    DT15    ;$ERRPC,$TMP0,$TMP1,0
1045  001316  067766                    DF15    ;0,C
1046
1047                                    ;ITEM 16
1048
1049  001320  067045                    EM16    ;CACHE DATA LOGGED INCORRECTLY
1050  001322  067617                    DH44    ;PC EXPCT RECVD
1051  001324  067700                    DT24    ;$ERRPC,$REG1,$REG0,0
1052  001326  000000                    0
1053
1054                                    ;ITEM 17
1055
1056  001330  067016                    EM45    ;CACHE TAG LOGGED WRONG
1057  001332  067617                    DH44    ;PC EXPCT RECVD
1058  001334  067700                    DT24    ;$ERRPC,$REG0,$REG1,0
1059  001336  000000                    0
1060
1061                                    ;ITEM 20
1062
1063  001340  065715                    EM26    ;PHYS. BA LOGGED WRONG
1064  001342  067617                    DH44    ;PC EXPCT RECVD
1065  001344  067700                    DT24    ;$ERRPC,$REG1,$REG0,0
1066  001346  000000                    0
1067
1068                                    ;ITEM 21
1069  001350  065545                    EM21    ;CSP CONSTANT WRONG
1070  001352  067346                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1071  001354  067650                    DT21    ;$ERRPC,$TMP1,$TMP2,$REG0,0
1072  001356  067770                    DF17    ;0,0,0
```

```
1073
1074                                    ;ITEM 22
1075  001360  065570                    EM22    ;BAD DATA READ BY A MED
1076  001362  067346                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1077  001364  067662                    DT22    ;$ERRPC,$TMP1,$TMP2,$TMP3,0
1078  001366  067770                    DF17    ;0,0,0
1079
1080                                    ;ITEM 23
1081  001370  065617                    EM23    ;NO ODD PC TRAP
1082  001372  067403                    DH23    ;PC
1083  001374  067674                    DT23    ;$ERRPC
1084  001376  000000                    0
1085
1086                                    ;ITEM 24
1087
1088  001400  065636                    EM24    ;ODD ADR. BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1089  001402  067410                    DH24    ;PC CPUERR LOGJAM
1090  001404  067700                    DT24    ;$ERRPC,$REG1,$REG0
1091  001406  000000                    0
1092
1093                                    ;ITEM 25
1094
1095  001410  065521                    EM17    ;LOG CUA LOGGED INCORRECT U-ADDR
1096  001412  067617                    DH44    ;PC     EXPCTD  RECVD
1097  001414  067700                    DT24    ;$ERRPC $REG1   $REG0
1098  001416  000000                    0
1099
1100                                    ;ITEM 26
1101
1102  001420  065715                    EM26    ;PHYS. BA LOGGED WRONG
1103  001422  067447                    DH26    ;PC PA<17:16>-EXPCT-PA<15:0>  PA<17:16>-RECVD-PA<15:0>
1104  001424  067716                    DT26    ;$ERRPC,$REG1,$REG2,$REG0,$REG3,0
1105  001426  000000                    0
1106
1107                                    ;ITEM 27
1108
1109  001430  065742                    EM27    ;CACHE PARITY ERROR LOGGED IN BACK UP MODE
1110  001432  067531                    DH27    ;PC LOGPBA LOGDATA LOGTAG
1111  001434  067732                    DT27    ;$ERRPC,$REG3,$REG1,$REG2
1112  001436  000000                    0
1113
1114                                    ;ITEM 30
1115
1116  001440  066012                    EM30    ;CACHE PARITY TRAPPED WHEN DISABLED
1117  001442  067403                    DH23    ;PC
1118  001444  067674                    DT23    ;$ERRPC
1119  001446  000000                    0
1120
1121                                    ;ITEM 31
1122
1123  001450  066615                    EM31    ;NO CACHE PARITY TRAP
1124  001452  067403                    DH23    ;PC
1125  001454  067674                    DT23    ;$ERRPC
1126  001456  000000                    0
1127
1128                                    ;ITEM 32
```

```
1129
1130  001460  066124                  EM32    ;MEMORY ERROR REGISTERS INCORRECT
1131  001462  067564                  DH32    ;PC MEMERR
1132  001464  067710                  DT25    ;$ERRPC,$REG0
1133  001466  000000                  0
1134
1135                                  ;ITEM 33
1136
1137  001470  066155                  EM33    ;TIMEOUT BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1138  001472  067410                  DH24    ;PC CPUERR LOGJAM
1139  001474  067700                  DT24    ;$ERRPC,$REG1,$REG0
1140  001476  000000                  0
1141
1142                                  ;ITEM 34
1143
1144  001500  066233                  EM34    ;NO ILLEGAL INTERNAL ADDRESS TRAP
1145  001502  067403                  DH23    ;PC
1146  001504  067674                  DT23    ;$ERRPC
1147  001506  000000                  0
1148
1149                                  ;ITEM 35
1150
1151  001510  066270                  EM35    ;INTERNAL ADDRESS ERROR BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1152  001512  067410                  DH24    ;PC CPUERR LOGJAM
1153  001514  067700                  DT24    ;$ERRPC,$REG1,$REG0
1154  001516  000000                  0
1155
1156                                  ;ITEM 36
1157
1158  001520  066356                  EM36    ;LAST INTERRUPT/TRAP VECTOR NOT LOGGED IN FLAG REGISTER
1159  001522  067433                  DH25    ;PC FLGREG
1160  001524  067710                  DT25    ;$ERRPC,$REG0
1161  001526  000000                  0
1162
1163                                  ;ITEM 37
1164
1165  001530  066433                  EM37    ;LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR
1166  001532  067410                  DH24    ;PC CPUERR LOGJAM
1167  001534  067700                  DT24    ;$ERRPC,$REG1,$REG0
1168  001536  000000                  0
1169
1170                                  ;ITEM 40
1171
1172  001540  066526                  EM40    ;ERROR LOG WAS NOT RE-ENABLED, ODD ADR BIT CLR IN CPUERR
1173  001542  067410                  DH24    ;PC CPUERR LOGJAM
1174  001544  067700                  DT24    ;$ERRPC,$REG1,$REG0
1175  001546  000000                  0
1176
1177                                  ;ITEM 41
1178
1179  001550  066055                  EM41    ;INSTRUCTION NOT ABORTED IN CACHE ABORT MODE
1180  001552  067403                  DH23    ;PC
1181  001554  067674                  DT23    ;$ERRPC
1182  001556  000000                  0
1183
1184                                  ;..EM 42
```

```
1185
1186  001560  066642                EM42    ;LO BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE
1187  001562  067600                DH42    ;PC LOGSERVCE
1188  001564  067710                DT25    ;$ERRPC,$REG0,0
1189  001566  000000                0
1190
1191                                ;ITEM 43
1192
1193  001570  066730                EM43    ;LO BYTE & TAG PARITY BITS NOT SET IN MEM ERR REGISTER
1194  001572  067564                DH32    ;PC MEMERR
1195  001574  067710                DT25    ;$ERRPC,$REG0
1196  001576  000000                0
1197
1198                                ;ITEM 44
1199
1200  001600  067075                EMEIS1  ;EIS SET COND CODES WRONG
1201  001602  067232                DHEIS1  ;  PSW    REG-WAS-REG+1    REG-S/B-REG+1      PC      TEST     (IR)
1202  001604  067744                DTEIS1  ;$REGAD $REG2    $REG3    $REG1    $REG4    $ERRPC  $REG0    $TMP0
1203  001606  000000                0
1204
1205                                ;ITEM 45
1206
1207  001610  067126                EMEIS2  ;EIS GAVE WRONG RESULT
1208  001612  067232                DHEIS1  ;  PSW    REG-WAS-REG+1    REG-S/B-REG+1      PC      TEST     (IR)
1209  001614  067744                DTEIS1  ;$REGAD $REG2    $REG3    $REG1    $REG4    $ERRPC  $REG0    $TMP0
1210  001616  000000                0
1211
1212                                ;ITEM 46
1213
1214  001620  067154                EM46    ;AUTO-INCREMENT (DECREMENT) DID NOT OCCUR
1215  001622  067273                DH46    ;  PC     (IR)     TEST
1216  001624  067756                DT46    ;$ERRPC $TMP0    $REG0
1217  001626  000000                0
1218
1219          076600                        MED = 076600
1220          140000                        UM=     140000
1221          177770                        UBREAK= 177770
1222          177744                        MEMERR=177744
1223          177766                        CPUERR=177766
1224          177746                        CCR=177746
1225          000100                        WWP=BIT6
1226          000001                        DPTRP=BIT0
1227          000200                        PABORT=BIT7
1228          000100                        LO=BIT6
1229          000200                        HI=BIT7
1230          000040                        TAG=BIT5
1231
1232                                        .EQUIV  SP,KSP
1233
1234
1235                                ;*      MED OPERATION CODE DEFINITIONS
1236
1237          000226                        WCNSSW=226
1238          000022                        RDWHAMI=022
1239          000222                        WRWHAMI=222
1240          000144                        RDFLAG=144
```

```
1241            000344                  WRFLAG=344
1242            000100                  RDLJAM=100
1243            000300                  WRLJAM=300
1244            000101                  RDLSERVICE=101
1245            000301                  WRLSERVICE=301
1246            000102                  RDLPBA=102
1247            000302                  WRLPBA=302
1248            000103                  RDLCUA=103
1249            000303                  WRLCUA=303
1250            000104                  RDLFGINT=104
1251            000304                  WRLFGINT=304
1252            000105                  RDLWHAMI=105
1253            000305                  WRLWHAMI=305
1254            000106                  RDLDATA=106
1255            000306                  WRLDATA=306
1256            000107                  RDLTAG=107
1257            000307                  WRLTAG=307
1258            000071                  SWB01=71                ;MICRO ADDR. IN SWAB INST.
1259
1260
1261
1262                                    ;ADDRESS ASSIGNMENTS FOR DL11 CONSOLE TERMINAL INTERFACE
1263
1264            177560                  RCSR=177560             ;RCVR. CONTROL / STATUS REG. ADDRESS
1265            177562                  RDBR = 177562           ;RECEIVER DATA BUFFER REG. ADDR.
1266            177564                  XCSR = 177564           ;TRANSMITTER CONTROL / STATUS REG. ADDR
1267            177566                  XDBR = 177566           ;TRANSMIT DATA BUFFER REG. ADDR.
1268            177546                  LKCSR=  177546          ;LINE CLOCK ADDRESS
1269
```

```
                                        ;//////////////////////
                                        ;  "BCPT" TESTS      /
                                        ;//////////////////////


                        ; ***************************************************************************
                        ;       .SBTTL BT001 "BR" TEST -POSITIVE OFFSET
                        ; ***************************************************************************

001630                  START:
001630  000401          BT001:  BR      BT002           ;TEST THE BR FORWARD

001632  000000          E001:   HALT                    ;BR FAILED TO LOAD PC PROPERLY

                        ; ***************************************************************************
                        ;       .SBTTL BT002 "BR" TEST - NEGATIVE OFFSET
                        ; ***************************************************************************

001634  000402          BT002:  BR      I002            ;GO TO TEST INSTRUCTION

001636  000403          A002:   BR      BT003           ;GO TO NEXT TEST

001640  000000          EX002:  HALT                    ;JUST IN CASE

001642  000775          I002:   BR      A002            ;TEST THE BR - NEG. OFFSET

001644  000000          E2002:  HALT                    ;BR FAILED WITH NEG. OFFSET

                        ; ***************************************************************************
                        ;       .SBTTL BT003 "BASIC COND. BR" TEST - FLAGS CLEARED
                        ; ***************************************************************************

001646  100403          BT003:  BMI     E003            ;BR IF "N" SET
001650  001402                  BEQ     E003            ;BR IF "Z" SET
001652  102401                  BVS     E003            ;BR IF "V" SET
001654  103002                  BCC     BT004           ;BR IF "C" CLEAR

001656  000000          E003:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
                                                        ;OR THE FLAGS FAILED TO CLEAR ON "START"
001660  000772                  BR      BT003           ;LOCK ON HARD ERROR

                        ; ***************************************************************************
                        ;       .SBTTL BT004 "SCC AND COND. BR'S" TEST - FLAGS SET
                        ; ***************************************************************************

001662  000277          BT004:  SCC                     ;MAKE N:C=1111

001664  100003          I004:   BPL     E004            ;BR IF "N" FAILED TO SET
001666  001002                  BNE     E004            ;BR IF "Z" FAILED TO SET
001670  102001                  BVC     E004            ;BR IF "V" FAILED TO SET
001672  103402                  BCS     BT005           ;BR IF "C" SET OK

001674  000000          E004:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
                                                        ;OR THA SCC FAILED TO SET ALL THE FLAGS
001676  000771                  BR      BT004           ;LOCK ON HARD ERROR

                        ; ***************************************************************************
```

```
1326                                            .SBTTL BTOOS "CCC AND COND. BR'S" TEST - FLAGS CLEARED
1327                                    ; ******************************************************************
1328
1329  001700  000257                   BTOOS:  CCC                     ;MAKE N:C=0000
1330
1331  001702  100403                   IOOS:   BMI     EOOS            ;BR IF "N" STILL SET
1332  001704  001402                           BEQ     EOOS            ;BR IF "Z" STILL SET
1333  001706  102401                           BVS     EOOS            ;BR IF "V" STILL SET
1334  001710  103002                           BCC     BTOO6           ;BR IF "C" GOT CLEARED
1335
1336  001712  000000                   EOOS:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
1337                                                                    ;OR THE CCC FAILED TO CLEAR ALL FLAGS
1338  001714  000771                           BR      BTOOS           ;LOCK ON HARD ERROR
1339
1340                                    ; ******************************************************************
1341                                            .SBTTL BTOO6 "CLR  %R" TEST - SETS THE "Z" BIT
1342                                    ; ******************************************************************
1343
1344  001716  000257                   BTOO6:  CCC                     ;MAKE N:C=0000
1345
1346  001720  005000                   IOO6:   CLR     R0              ;TEST THE CLR - IT SHOULD SET "Z"
1347
1348  001722  001402                           BEQ     BTOO7           ;BR IF CLR SET "Z"
1349
1350  001724  000000                   EOO6:   HALT                    ;ERROR - CLR FAILED TO SET "Z"
1351  001726  000773                           BR      BTOO6           ;LOCK ON HARD ERROR
1352
1353                                    ; ******************************************************************
1354                                            .SBTTL BTOO7 "TST  %R" TEST - USING THE CLR
1355                                    ; ******************************************************************
1356
1357  001730  005000                   BTOO7:  CLR     R0              ;MAKE (R0) = 000000
1358  001732  000257                           CCC                     ;MAKE N:C=0000
1359
1360  001734  005700                   IOO7:   TST     R0              ;TEST THE TST - IT SHOULD SET "Z"
1361
1362  001736  001402                           BEQ     BTO10           ;BR IF "Z" SET OK
1363
1364  001740  000000                   EOO7:   HALT                    ;ERROR - CLR FAILED TO LOAD R0 WITH
1365                                                                    ;ALL ZEROES OR TST FAILED
1366  001742  000772                           BR      BTOO7           ;LOCK ON HARD ERROR
1367
1368                                    ; ******************************************************************
1369                                            .SBTTL BTO10 "COM  %R" TEST - SHOULD SET "N" AND "C"
1370                                    ; ******************************************************************
1371
1372  001744  005000                   BTO10:  CLR     R0              ;MAKE (R0) = 000000
1373  001746  000257                           CCC                     ;MAKE N:C=0000
1374
1375  001750  005100                   IO10:   COM     R0              ;TEST THE COM - (R0) S/B = 177777
1376
1377  001752  100001                           BPL     E010            ;BR IF "N" FAILED TO SET
1378  001754  103402                           BCS     BTO11           ;BR IF "C" SET OK
1379
1380  001756  000000                   E010:   HALT                    ;ERROR - COM FAILED
1381  001760  000771                           BR      BTO10           ;LOCK ON HARD ERROR
```

```
1382
1383                            ; ***************************************************************************
1384                            ;       .SBTTL BTO11 "COM  %R AND ADC  %R" TEST
1385                            ; ***************************************************************************
1386
1387  001762  005000            BTO11:  CLR     R0              ;MAKE [R0] = 000000
1388  001764  000257                    CCC                     ;MAKE N:C=0000
1389
1390  001766  005100            IO11:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
1391  001770  005500                    ADC     R0              ;TEST THE ADC - [R0] S/B = 000000
1392
1393  001772  001001                    BNE     EO11            ;BR IF "Z" DID NOT SET
1394  001774  103402                    BCS     BTO12           ;BR IF "C" SET OK
1395
1396  001776  000000            EO11:   HALT                    ;ERROR - COM OR ADC FAILED
1397  002000  000770                    BR      BTO11           ;LOCK ON HARD ERROR
1398
1399                            ; ***************************************************************************
1400                            ;       .SBTTL BTO12 "MOV  #N,R" TEST WITH N=177777,[R]=000000
1401                            ; ***************************************************************************
1402
1403  002002  005000            BTO12:  CLR     R0              ;MAKE [R0] = 000000
1404  002004  000257                    CCC                     ;MAKE N:C=0000
1405
1406  002006  012700  177777    IO12:   MOV     #-1,R0          ;TEST THE MOV - [R0] S/B = 177777
1407
1408  002012  005100                    COM     R0              ;MAKE [R0] = 000000
1409  002014  001402                    BEQ     BTO13           ;BR IF "Z" SET
1410
1411  002016  000000            EO12:   HALT                    ;ERROR - MOV FAILED TO LOAD R0 WITH ALL 1'S
1412  002020  000770                    BR      BTO12           ;LOCK ON HARD ERROR
1413
1414                            ; ***************************************************************************
1415                            ;       .SBTTL BTO13 "MOV  #N,R" TEST WITH N=000000,[R]=177777
1416                            ; ***************************************************************************
1417
1418  002022  005000            BTO13:  CLR     R0              ;MAKE [R0] = 000000
1419  002024  005100                    COM     R0              ;MAKE [R0] = 177777
1420  002026  000257                    CCC                     ;SCOPE SYNC
1421
1422  002030  012700  000000    IO13:   MOV     #0,R0           ;TEST THE MOV - [R0] S/B = 000000
1423
1424  002034  005100                    COM     R0              ;MAKE [R0] = 177777, SET "C"
1425  002036  005500                    ADC     R0              ;MAKE [R0] = 000000
1426  002040  001402                    BEQ     BTO14           ;BR IF "Z" GOT SET
1427
1428  002042  000000            EO13:   HALT                    ;ERROR - MOV FAILED TO CLEAR R0
1429  002044  000766                    BR      BTO13           ;LOCK ON HARD ERROR
1430
1431                            ; ***************************************************************************
1432                            ;       .SBTTL BTO14 "CLR  (R)" TEST - [R] = 177776
1433                            ; ***************************************************************************
1434
1435  002046  012706  001000    BTO14:  MOV     #STACK,SP       ;SET UP STACK POINTER
1436  002052  012700  177776            MOV     #PSW,R0         ;R0 POINTS TO PSW
1437  002056  000277                    SCC                     ;MAKE [PSW] = 017
```

```
                                ;TEST THE CLR - IT SHOULD CLEAR PSW
002060  005010          IO14:   CLR     (R0)
                                ;BR IF CLR MADE "Z" = 0 - IT SHOULD
002062  001002                  BNE     BTO15
                                ;ERROR- CLR FAILED TO CLEAR PSW
002064  000000          EO14:   HALT
                                ;LOCK ON HARD ERROR
002066  000767                  BR      BTO14

                        ; ***************************************************************************
                                .SBTTL BTO15 "CLR (R)+" TEST - [R] = 177776
                        ; ***************************************************************************

                                ;R0 POINTS TO PSW
002070  012700  177776  BTO15:  MOV     #PSW,R0
                                ;MAKE [PSW] = 017
002074  000277                  SCC

                                ;TEST THE CLR - IT SHOULD CLEAR PSW
002076  005020          IO15:   CLR     (R0)+
                                ;BR IF CLR MADE "Z" = 0 - IT SHOULD
002100  001002                  BNE     AO15
                                ;ERROR- CLR FAILED TO CLEAR PSW
002102  000000          E1015A: HALT
                                ;LOCK ON HARD ERROR
002104  000771                  BR      BTO15

                                ;AUTO INC SHOULD ZERO R0
002106  005700          AO15:   TST     R0
                                ;BR IF IT DID
002110  001402                  BEQ     BTO16
                                ;ERROR - AUTOINC. FAILED
002112  000000          E2015:  HALT
                                ;LOCK ON HARD ERROR
002114  000765                  BR      BTO15

                        ; ***************************************************************************
                                .SBTTL BTO16 "COM (R)" TEST - [R] = 177776
                        ; ***************************************************************************

                                ;R0 POINTS TO PSW
002116  012700  177776  BTO16:  MOV     #PSW,R0
                                ;MAKE [PSW] = 000
002122  000257                  CCC

                                ;TEST THE COM - [PSW] S/B = 357
002124  005110          IO16:   COM     (R0)

002126  100003                  BPL     EO16            ;N:C=1111 ?
002130  001002                  BNE     EO16
002132  102001                  BVC     EO16
002134  103403                  BCS     BTO17

                                ;GO TO KERNEL MODE
002136  005010          EO16:   CLR     (R0)
                                ;ERROR - COM FAILED TO MAKE [PSW] = 357
002140  000000                  HALT
                                ;LOCK ON HARD ERROR
002142  000765                  BR      BTO16

                        ; ***************************************************************************
                                .SBTTL BTO17 "COM (R0)+" TEST - [R0] = 177776
                        ; ***************************************************************************

                                ;R0 POINTS TO PSW
002144  012700  177776  BTO17:  MOV     #PSW,R0
                                ;MAKE [PSW] = 000
002150  005010                  CLR     (R0)
                                ;SCOPE SYNC
002152  000257                  CCC

                                ;TEST THE COM - [PSW] S/B = 357
002154  005120          IO17:   COM     (R0)+
```

```
1494
1495  002156  100003                    BPL     EA017           ;N:C = 1111 ?
1496  002160  001002                    BNE     EA017
1497  002162  102001                    BVC     EA017
1498  002164  103405                    BCS     A017
1499
1500  002166  012701  177776  EA017:    MOV     #PSW,R1
1501  002172  005011                    CLR     (R1)
1502  002174  000000                    HALT                    ;COM FAILED TO SET ALL FLAGS
1503  002176  000762                    BR      BTO17           ;LOCK ON HARD ERROR
1504
1505  002200  005100          A017:     COM     R0              ;SHOULD MAKE [R0] = 177777
1506  002202  005500                    ADC     R0              ;SHOULD MAKE [R0] = 000000
1507  002204  001405                    BEQ     BTO20
1508
1509  002206  012701  177776  E2017:    MOV     #PSW,R1
1510  002212  005011                    CLR     (R1)
1511  002214  000000                    HALT                    ;ERROR - COM FAILED TO AUTO INC. R0
1512  002216  000752                    BR      BTO17           ;LOCK ON HARD ERROR
1513
1514                        ; ***************************************************************************
1515                                .SBTTL BTO20 "MOV  RA,RB" TEST - WITH [RA]=177777,[RB]=000000
1516                        ; ***************************************************************************
1517
1518  002220  012700  177776  BTO20:   MOV     #PSW,R0
1519  002224  005010                    CLR     (R0)
1520  002226  005000                    CLR     R0              ;MAKE [R0]=000000
1521  002230  005001                    CLR     R1              ;MAKE [R1]=000000
1522  002232  005101                    COM     R1              ;MAKE [R1]=0207777
1523  002234  000257                    CCC                     ;SCOPE SYNC
1524
1525  002236  010100          I020:     MOV     R1,R0           ;TEST THE MOV
1526
1527  002240  100402                    BMI     A020            ;BR IF "N" GOT SET
1528
1529  002242  000000          EA020:    HALT                    ;ERROR-MOV FAILED TO SET "N"
1530  002244  000765                    BR      BTO20           ;LOCK ON HARD ERROR
1531
1532  002246  005100          A020:     COM     R0              ;[R0] SHOULD GO TO 000000
1533  002250  001402                    BEQ     BTO21           ;BR IF IT DID
1534
1535  002252  000000          E2020:    HALT                    ;ERROR-MOV FAILED TO LOAD R0 WITH 1'S
1536  002254  000761                    BR      BTO20           ;LOCK ON HARD ERROR
1537
1538                        ; ***************************************************************************
1539                                .SBTTL BTO21 "MOV  RA,RB" TEST WITH [RA]=000000,[RB]=177777
1540                        ; ***************************************************************************
1541
1542  002256  005000          BTO21:    CLR     R0              ;MAKE [R0]=000000
1543  002260  005100                    COM     R0              ;MAKE [R0]=177777
1544  002262  005001                    CLR     R1              ;MAKE [R1]=000000
1545  002264  000257                    CCC                     ;SCOPE SYNC
1546
1547  002266  010100          I021:     MOV     R1,R0           ;TEST THE MOV
1548
1549  002270  001402                    BEQ     A021            ;BR IF "Z" GOT SET
```

```
1550
1551  002272  000000                  EA021:  HALT                    ;MOV FAILED TO SET "Z"
1552  002274  000770                          BR      BT021           ;LOCK ON HARD ERROR
1553
1554  002276  005100                  A021:   COM     R0              ;SHOULD MAKE [R0]=177777 AND SET "C"
1555  002300  005500                          ADC     R0              ;SHOULD MAKE [R0]=000000
1556  002302  001402                          BEQ     BT022           ;BR IF "Z" SET
1557
1558  002304  000000                  E2021:  HALT                    ;MOV FAILED TO ZERO R0
1559  002306  000763                          BR      BT021           ;LOCK ON HARD ERROR
1560
1561                                  ; ***************************************************************************
1562                                          .SBTTL BT022 "MOV  #N,@#A" TEST WITH N=17,A=177776
1563                                  ; ***************************************************************************
1564
1565  002310  000257                  BT022:  CCC                     ;MAKE [PSW]=000
1566
1567  002312  012737  000017  177776  I022:   MOV     #17,@#PSW       ;TEST THE MOV
1568
1569  002320  100003                          BPL     E022            ;N:C=1111
1570  002322  001002                          BNE     E022
1571  002324  102001                          BVC     E022
1572  002326  103402                          BCS     BT023
1573
1574  002330  000000                  E022:   HALT                    ;MOV FAILED TO LOAD PSW
1575  002332  000766                          BR      BT022           ;LOCK ON HARD ERROR
1576
1577                                  ; ***************************************************************************
1578                                          .SBTTL BT023 "MOV  RA,(RB)+" TEST WITH [RA]=17,[RB]=177776
1579                                  ; ***************************************************************************
1580
1581  002334  012700  177776          BT023:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1582  002340  012701  000017                  MOV     #17,R1          ;[SOURCE]=017
1583  002344  000257                          CCC                     ;SCOPE SYNC - MAKE <N:C> = 0000
1584
1585  002346  010120                  I023:   MOV     R1,(R0)+        ;TEST THE MOV
1586
1587  002350  100003                          BPL     EA023           ;N:C = 1111 ?
1588  002352  001002                          BNE     EA023
1589  002354  102001                          BVC     EA023
1590  002356  103402                          BCS     A023
1591
1592  002360  000000                  EA023:  HALT                    ;MOV FAILED TO LOAD PSW
1593  002362  000764                          BR      BT023           ;LOCK ON HARD ERROR
1594
1595  002364  005700                  A023:   TST     R0              ;DID AUTO INC MAKE R0 GO TO 0?
1596  002366  001402                          BEQ     BT024           ;BR IF IT DID
1597
1598  002370  000000                  E2023:  HALT                    ;MOV FAILED TO AUTO INC. R0
1599  002372  000760                          BR      BT023           ;LOCK ON HARD ERROR
1600
1601                                  ; ***************************************************************************
1602                                          .SBTTL BT024 "CMP  #N,@#A" TEST WITH N=(A)
1603                                  ; ***************************************************************************
1604
1605  002374  012700  177776          BT024:  MOV     #PSW,R0         ;R0 POINTS TO PSW
```

```
1606  002400  005010                        CLR     (R0)            ;MAKE [PSW]=000
1607  002402  000273                        273                     ;MAKE N:C=1011
1608
1609  002404  022737  000013  177776  I024: CMP     #13,@#PSW       ;TEST THE CMP
1610
1611  002412  001402                        BEQ     BT025           ;BR IF "Z" GOT SET
1612
1613  002414  000000                E024:   HALT                    ;CMP FAILED TO SET "Z"
1614  002416  000766                        BR      BT024           ;LOCK ON HARD ERROR
1615
1616                                ; ***************************************************************************
1617                                        .SBTTL BT025 "CMP  #N,@#A" WITH N > (A)
1618                                ; ***************************************************************************
1619
1620  002420  000257                BT025:  CCC                     ;MAKE [PSW]=000
1621
1622  002422  022737  000017  177776  I025: CMP     #17,@#PSW       ;TEST THE CMP
1623
1624  002430  001401                        BEQ     E025            ;BR IF "Z" GOT SET
1625  002432  000402                        BR      BT026           ;GO TO NEXT TEST
1626
1627  002434  000000                E025:   HALT                    ;CMP FAILED TO CLEAR "Z"
1628  002436  000770                        BR      BT025           ;LOCK ON HARD ERROR
1629                                ; ***************************************************************************
1630                                        .SBTTL BT026 "CMP  #N,@#A" WITH N < (A)
1631                                ; ***************************************************************************
1632
1633  002440  000277                BT026:  SCC                     ;MAKE [PSW]=017
1634
1635  002442  022737  000000  177776  I026: CMP     #0,@#PSW        ;TEST THE CMP
1636
1637  002450  001401                        BEQ     E026            ;BR IF "Z" GOT SET
1638  002452  000402                        BR      BT027           ;GO TO NEXT TEST
1639
1640  002454  000000                E026:   HALT                    ;CMP FAILED TO CLEAR "Z"
1641  002456  000770                        BR      BT026           ;LOCK ON HARD ERROR
1642
1643                                ; ***************************************************************************
1644                                        .SBTTL BT027 "CMP  R,#N" TEST WITH [R]=N
1645                                ; ***************************************************************************
1646
1647  002460  012700  177777        BT027:  MOV     #-1,R0          ;MAKE [R0]=177777
1648  002464  000257                        CCC                     ;N:C=0000
1649
1650  002466  020027  177777        I027:   CMP     R0,#-1          ;TEST THE CMP
1651
1652  002472  001402                        BEQ     BT030           ;BR IF CMP SET "Z"
1653
1654  002474  000000                E027:   HALT                    ;CMP FAILED
1655  002476  000770                        BR      BT027           ;LOCK ON HARD ERROR
1656
1657                                ; ***************************************************************************
1658                                        .SBTTL BT030 "CMP  R,#N" TEST WITH [R] > N
1659                                ; ***************************************************************************
1660
1661  002500  012700  000001        BT030:  MOV     #1,R0           ;MAKE [R0]=000001
```

```
1662  002504  000264                          SEZ                     ;SET THE "Z" BIT
1663
1664  002506  020027  177777        I030:     CMP     R0,#-1          ;TEST THE CMP
1665
1666  002512  001002                          BNE     BT031           ;BR IF CMP CLEARED "Z"
1667
1668  002514  000000        E030:     HALT                            ;CMP FAILED
1669  002516  000770                          BR      BT030           ;LOCK ON HARD ERROR
1670                        ; ***************************************************************
1671                                .SBTTL BT031 "CMP  R,#N" TEST WITH [R] < N
1672                        ; ***************************************************************
1673
1674  002520  012700  000001    BT031:  MOV     #1,R0           ;MAKE [R0] = 000001
1675  002524  000264                          SEZ                     ;SET THE "Z" BIT
1676
1677  002526  020027  000017        I031:     CMP     R0,#17          ;TEST THE CMP
1678
1679  002532  001002                          BNE     BT032           ;BR IF CMP CLEARED "Z"
1680
1681  002534  000000        E031:     HALT                            ;CMP FAILED TO SET "Z"
1682  002536  000770                          BR      BT031           ;LOCK ON HARD ERROR
1683
1684                        ; ***************************************************************
1685                                .SBTTL BT032 "CMP  (R0)+,R0" TEST WITH [SOURCE]=[R0]
1686                        ; ***************************************************************
1687
1688  002540  012700  177776    BT032:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1689  002544  012737  000340  177776    MOV     #340,@#PSW      ;MAKE [PSW]=340
1690  002552  012701  000340            MOV     #340,R1         ;MAKE [DEST]=340
1691  002556  000257                    CCC                     ;N:C=0000
1692
1693  002560  022001        I032:     CMP     (R0)+,R1        ;TEST THE CMP
1694
1695  002562  001402                          BEQ     A032            ;BR IF "Z" GOT SET
1696
1697  002564  000000        EA032:    HALT                            ;CMP FAILED TO ACCESS PSW
1698  002566  000764                          BR      BT032           ;LOCK ON HARD ERROR
1699
1700  002570  005700        A032:     TST     R0              ;"Z" SHOULD SET
1701  002572  001402                          BEQ     BT033           ;BR IF "Z" SET
1702
1703  002574  000000        E2032:    HALT                            ;CMP FAILED TO AUTO INC. R0
1704  002576  000760                          BR      BT032           ;LOCK ON HARD ERROR
1705
1706                        ; ***************************************************************
1707                                .SBTTL BT033 "CMP  (R0)+,R0" TEST WITH [SOURCE]>[R0]
1708                        ; ***************************************************************
1709
1710  002600  012700  177776    BT033:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1711  002604  012737  000340  177776    MOV     #340,@#PSW      ;MAKE [PSW]=340
1712  002612  012701  000330            MOV     #330,R1         ;MAKE [DEST]=330
1713  002616  000264                    SEZ                     ;SET THE "Z" BIT
1714
1715  002620  022001        I033:     CMP     (R0)+,R1        ;TEST THE CMP
1716
1717  002622  001002                          BNE     A033            ;BR IF "Z" GOT CLEARED
```

```
1718
1719  002624  000000                  EA033:  HALT                    ;CMP FAILED TO ACCESS PSW
1720  002626  000764                          BR      BT033           ;LOCK ON HARD ERROR
1721
1722  002630  005700                  A033:   TST     R0              ;"Z" SHOULD SET
1723  002632  001402                          BEQ     BT034           ;BR IF "Z" SET
1724
1725  002634  000000                  E2033:  HALT                    ;CMP FAILED TO AUTO INC. R0
1726  002636  000760                          BR      BT033           ;LOCK ON HARD ERROR
1727                                  ; ***************************************************************
1728                                          .SBTTL BT034 "CMP (RA)+,RB" TEST WITH [SOURCE]<[RB]
1729                                  ; ***************************************************************
1730
1731  002640  012700  177776          BT034:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1732  002644  012737  000330  177776          MOV     #330,@#PSW      ;MAKE [PSW]=330
1733  002652  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
1734  002656  000264                          SEZ                     ;SET THE "Z" BIT
1735
1736  002660  022001                  I034:   CMP     (R0)+,R1        ;TEST THE CMP
1737
1738  002662  001002                          BNE     A034            ;BR IF "Z" GOT CLEARED
1739
1740  002664  000000                  EA034:  HALT                    ;CMP FAILED TO ACCESS PSW
1741  002666  000764                          BR      BT034           ;LOCK ON HARD ERROR
1742
1743  002670  005700                  A034:   TST     R0              ;"Z" SHOULD SET
1744  002672  001402                          BEQ     BT035           ;BR IF "Z" SET
1745
1746  002674  000000                  E2034:  HALT                    ;CMP FAILED TO AUTO INC. R0
1747  002676  000760                          BR      BT034           ;LOCK ON HARD ERROR
1748                                  ; ***************************************************************
1749                                          .SBTTL BT035 "CMP RA,RB" TEST WITH [RA] = [RB]
1750                                  ; ***************************************************************
1751
1752  002700  012700  125252          BT035:  MOV     #125252,R0      ;MAKE [R0] = 125252
1753  002704  010001                          MOV     R0,R1           ;MAKE [R1] = 125252
1754  002706  000257                          CCC                     ;SCOPE SYNC
1755
1756  002710  020100                  I035:   CMP     R1,R0           ;TEST THE CMP
1757
1758  002712  001402                          BEQ     BT036           ;BR IF "Z" GOT SET
1759
1760  002714  000000                  E035:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1761  002716  000770                          BR      BT035           ;LOCK ON HARD ERROR
1762                                  ; ***************************************************************
1763                                          .SBTTL BT036 "CMP RA,RB" TEST WITH [RA] < [RB]
1764                                  ; ***************************************************************
1765
1766  002720  012700  025252          BT036:  MOV     #25252,R0       ;MAKE [R0] = 25252
1767  002724  005001                          CLR     R1              ;MAKE [R1] = 000000
1768  002726  000264                          SEZ                     ;SCOPE SYNC - SET "Z"
1769
1770  002730  020100                  I036:   CMP     R1,R0           ;TEST THE CMP
1771
1772  002732  001002                          BNE     BT037           ;BR IF "Z" GOT  CLEARED
1773
```

```
1774  002734  000000                  E036:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1775  002736  000770                          BR      BT036           ;LOCK ON HARD ERROR
1776                                  ; ***************************************************************
1777                                          .SBTTL BT037 "CMP  RA,RB" TEST WITH [RA] > [RB]
1778                                  ; ***************************************************************
1779
1780  002740  005000                  BT037:  CLR     R0              ;MAKE [R0] = 000000
1781  002742  012701  000017                  MOV     #17,R1          ;MAKE [R1] = 000017
1782  002746  000264                          SEZ                     ;SCOPE SYNC - SET "Z"
1783
1784  002750  020100                  I037:   CMP     R1,R0           ;TEST THE CMP
1785
1786  002752  001002                          BNE     BT040           ;BR IF "Z" GOT CLEARED
1787
1788  002754  000000                  E037:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1789  002756  000770                          BR      BT037           ;LOCK ON HARD ERROR
1790
1791                                  ; ***************************************************************
1792                                          .SBTTL BT040 "MOV  (RA),RB" TEST WITH [SOURCE]=[RB]=17
1793                                  ; ***************************************************************
1794
1795  002760  012700  177776          BT040:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1796  002764  005010                          CLR     (R0)            ;MAKE [PSW]=000
1797  002766  005001                          CLR     R1              ;MAKE [R1]=000000
1798  002770  000277                          SCC                     ;MAKE N:C=1111
1799
1800  002772  011001                  I040:   MOV     (R0),R1         ;TEST THE MOV
1801
1802  002774  020127  000017                  CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1803  003000  001402                          BEQ     BT041           ;BR IF YES
1804
1805  003002  000000                  E040:   HALT                    ;MOV FAILED TO LOAD R1
1806  003004  000765                          BR      BT040           ;LOCK ON HARD ERROR
1807                                  ; ***************************************************************
1808                                          .SBTTL BT041 "MOV  (RA)+,RB" TEST WITH [SOURCE]=[RB]=17
1809                                  ; ***************************************************************
1810
1811  003006  012700  177776          BT041:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1812  003012  005010                          CLR     (R0)            ;MAKE [PSW]=000
1813  003014  005001                          CLR     R1              ;MAKE [R1]=000000
1814  003016  000277                          SCC                     ;MAKE N:C=1111
1815
1816  003020  012001                  I041:   MOV     (R0)+,R1                ;TEST THE MOV
1817
1818  003022  020127  000017                  CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1819  003026  001402                          BEQ     A041            ;BR IF YES
1820
1821  003030  000000                  EA041:  HALT                    ;MOV FAILED TO LOAD R1
1822  003032  000765                          BR      BT041           ;LOCK ON HARD ERROR
1823
1824  003034  005700                  A041:   TST     R0              ;"Z" SHOULD SET
1825  003036  001402                          BEQ     BT042           ;BR IF "Z" GOT SET
1826
1827  003040  000000                  E2041:  HALT                    ;MOV FAILED TO AUTO INC. R0
1828  003042  000761                          BR      BT041           ;LOCK ON HARD ERROR
1829
```

```
                        ; ************************************************************************
                        ;       .SBTTL BTO42 "XOR  RA,RB" TEST WITH [RA] = [RB] = 000000
                        ; ************************************************************************

003044  005000  BTO42:  CLR     R0              ;MAKE [R0] = 000000
003046  005001         CLR     R1              ;MAKE [R1] = 000000
003050  000257         CCC                     ;SCOPE SYNC

003052  074100  IO42:   XOR     R1,R0           ;TEST THE XOR

003054  005700         TST     R0              ;RESULT = 000000 ?
003056  001402         BEQ     BTO43           ;BR IF YES

003060  000000  EO42:   HALT                    ;XOR FAILED
003062  000770         BR      BTO42

                        ; ************************************************************************
                        ;       .SBTTL BTO43 "XOR  RA,RB" TEST WITH [RA] = [RB] = 177777
                        ; ************************************************************************

003064  005000  BTO43:  CLR     R0              ;MAKE [R0] = 177777
003066  005100         COM     R0
003070  010001         MOV     R0,R1           ;MAKE [R1] = 177777
003072  000257         CCC                     ;SCOPE SYNC

003074  074100  IO43:   XOR     R1,R0           ;TEST THE XOR

003076  005700         TST     R0              ;RESULT = 000000 ?
003100  001402         BEQ     BTO44   ;BR IF YES

003102  000000  EO43:   HALT                    ;XOR FAILED
003104  000767         BR      BTO43           ;LOCK ON HARD ERROR

                        ; ************************************************************************
                        ;       .SBTTL BTO44 "XOR  RA,RB" TEST WITH [RB]=052525,[RA]=125252
                        ; ************************************************************************

003106  012701  125252  BTO44:  MOV     #125252,R1      ;MAKE [R1]=125252
003112  012700  052525         MOV     #052525,R0      ;MAKE [R0]=052525
003116  000257         CCC                     ;SCOPE SYNC

003120  074100  IO44:   XOR     R1,R0           ;TEST THE XOR

003122  020027  177777         CMP     R0,#-1          ;RESULT = 177777 ?
003126  001402         BEQ     BTO45           ;BR IF YES

003130  000000  EO44:   HALT                    ;XOR FAILED
003132  000400         BR      BTO45           ;LOCK ON HARD ERROR
                        ; ************************************************************************
                        ;       .SBTTL BTO45 "XOR  RA,RB" TEST WITH [RA]=052525,[RB]=125252
                        ; ************************************************************************

003134  012700  125252  BTO45:  MOV     #125252,R0      ;MAKE [R0]=125252
003140  012701  052525         MOV     #052525,R1      ;MAKE [R1]=052525
003144  000257         CCC                     ;SCOPE SYNC
```

```
1886  003146  074100                  IO45:   XOR     R1,R0           ;TEST THE XOR
1887
1888  003150  020027  177777                  CMP     R0,#-1          ;RESULT = 177777 ?
1889  003154  001402                          BEQ     BTO46           ;BR IF YES
1890
1891  003156  000000                  EO45:   HALT                    ;XOR FAILED
1892  003160  000765                          BR      BTO45           ;LOCK ON HARD ERROR
1893
1894                                  ; ***************************************************************
1895                                          .SBTTL BTO46 GPR ADDRESS INTERRACTION TEST
1896                                  ; ***************************************************************
1897
1898  003162  012700  125252          BTO46:  MOV     #125252,R0      ;[R0] = 125252
1899  003166  010001                          MOV     R0,R1
1900  003170  005101                          COM     R1              ;[R1] = 052525
1901  003172  010102                          MOV     R1,R2
1902  003174  005102                          COM     R2              ;[R2] = 125252
1903  003176  010203                          MOV     R2,R3
1904  003200  005103                          COM     R3              ;[R3] = 052525
1905  003202  010304                          MOV     R3,R4
1906  003204  005104                          COM     R4              ;[R4] = 125252
1907  003206  010405                          MOV     R4,R5
1908  003210  005105                          COM     R5              ;[R5] = 052525
1909
1910  003212  074100                  IO46:   XOR     R1,R0           ;[R0] S/B = 177777
1911  003214  074200                          XOR     R2,R0           ;[R0] S/B = 125252
1912  003216  074300                          XOR     R3,R0           ;[R0] S/B = 177777
1913  003220  074400                          XOR     R4,R0           ;[R0]  S/B = 125252
1914  003222  074500                          XOR     R5,R0           ;[R0]  S/B = 177777
1915  003224  005100                          COM     R0              ;[R0] S/B = 000000
1916
1917  003226  001402                          BEQ     AO46            ;BR IF [R0] WAS 000000
1918
1919  003230  000000                  EAO46:  HALT                    ;GPR ADDRESSING PROBLEM
1920  003232  000753                          BR      BTO46           ;LOCK ON HARD ERROR
1921
1922  003234  020627  001000          AO46:   CMP     SP,#STACK       ;DID R6 GET DISTURBED
1923  003240  001402                          BEQ     BASIC           ;BR IF NOT
1924
1925  003242  000000                  E2O46:  HALT                    ;R6 ADDRESS PROBLEM
1926  003244  000746                          BR      BTO46           ;LOCK ON HARD ERROR
```

```
1927                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1928                                    ;        / / / / / / / BASIC INSTRUCTION TESTS / / / / / / /
1929                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1930
1931  003246  005037  063254            BASIC:  CLR     @#ONCE          ;SIGNAL PROGRAM HEADER TO BE PRINTED
1932  003252  005037  001012                    CLR     @#ERTTL         ;CLEAR ERROR COUNT FIRST TIME THROUGH
1933  003256  005037  001126                    CLR     @#SPASS         ;CLEAR PASS COUNT FIRST TIME THROUGH
1934  003262  012701  063236            INIT:   MOV     #PRIFLG,R1      ;SET UP TO INIT. COUNTERS AND FLAGS
1935  003266  005021                    1$:     CLR     (R1)+           ;CLEAR ONE WORD
1936  003270  020127  063254                    CMP     R1,#ONCE        ;CLEARED ALL FLAGS AND COUNTERS?
1937  003274  001374                            BNE     1$              ;BR IF NOT
1938  003276  012706  001000                    MOV     #STACK,SP       ;SET UP THE STACK POINTER
1939  003302  012737  004030  177770            MOV     #4030,@#UBREAK  ;SET SCOPE SYNC FOR COND CODE OPERATE
1940  003310  012737  177777  001074            MOV     #-1,@#SREGS     ;FLAG CURRENT STACK POINTER TO BE TYPED
1941                                                                    ;IN FIRST ERROR CALL
1942
1943                                    ;;**************************************************************
1944                                    ;*TEST 0        BASIC "BNE" TEST WITH Z=0
1945                                    ;;**************************************************************
1946  003316                            TST0:
1947  003316  012700  000000                    MOV     #0,R0           ;;LOAD R0 WITH TEST NUMBER
1948  003322  000257                    1$:     CCC                     ;MAKE Z=0
1949
1950  003324                            2$:             /
1951  003324  001002                            BNE     TST1            ;;TEST THE BNE - IT SHOULD BR
1952
1953  003326  000000                    3$:     HALT                    ;BNE FAILED TO LOAD PC
1954  003330  000774                            BR      1$              ;LOCK ON HARD ERROR
1955
1956                                    ;;**************************************************************
1957                                    ;*TEST 1        BASIC "BNE" TEST WITH Z=1
1958                                    ;;**************************************************************
1959  003332                            TST1:
1960  003332  012700  000001                    MOV     #1,R0           ;;LOAD R0 WITH TEST NUMBER
1961  003336  000264                    1$:     SEZ                     ;SET THE "Z" BIT
1962
1963  003340  001001                    2$:     BNE     3$              ;TEST THE BNE - IT SHOULD NOT BR
1964
1965  003342  000402                            BR      TST2            ;;GO TO NEXT TEST
1966
1967  003344  000000                    3$:     HALT                    ;BNE BRANCHED WITH Z=1
1968  003346  000773                            BR      1$              ;LOCK ON HARD ERROR
1969
1970                                    ;;**************************************************************
1971                                    ;*TEST 2        BASIC "BEQ" TEST WITH Z=1
1972                                    ;;**************************************************************
1973  003350                            TST2:
1974  003350  012700  000002                    MOV     #2,R0           ;;LOAD R0 WITH TEST NUMBER
1975  003354  000264                    1$:     SEZ                     ;MAKE Z=1
1976
1977  003356                            2$:
1978  003356  001402                            BEQ     TST3            ;;TEST THE BEQ - IT SHOULD BR
1979
1980  003360  000000                    3$:     HALT                    ;BEQ FAILED TO LOAD THE PC
1981  003362  000774                            BR      1$              ;LOCK ON HARD ERROR
1982                                    ;;**************************************************************
```

```
                                ;*TEST 3        BASIC "BEQ" TEST WITH Z=0
                                ;;**************************************************************
                                TST3:
003364                                  MOV     #3,R0           ;;LOAD R0 WITH TEST NUMBER
003364  012700  000003          1$:     CCC                     ;MAKE Z=0
003370  000257
                                2$:     BEQ     3$              ;TEST THE BEQ - IT SHOULD NOT BR
003372  001401
                                        BR      TST4            ;;GO TO NEXT TEST
003374  000402
                                3$:     HALT                    ;BEQ BRANCHED WITH Z=0
003376  000000                          BR      1$              ;LOCK ON HARD ERROR
003400  000773
                                ;;**************************************************************
                                ;*TEST 4        BASIC "BPL" TEST WITH N=1
                                ;;**************************************************************
                                TST4:
003402                                  MOV     #4,R0           ;;LOAD R0 WITH TEST NUMBER
003402  012700  000004          1$:     CLR     @#PSW           ;CLEAR THE PSW
003406  005037  177776                  SEN                     ;MAKE N=1
003412  000270
                                2$:     BPL     3$              ;TEST THE BPL - IT SHOULDN'T BR
003414  100001
                                        BR      TST5            ;;GO TO NEXT TEST
003416  000402
                                3$:     HALT                    ;BPL BRANCHED WITH N=1
003420  000000                          BR      1$              ;LOCK ON HARD ERROR
003422  000771
                                ;;**************************************************************
                                ;*TEST 5        BASIC "BPL" TEST WITH N=0
                                ;;**************************************************************
                                TST5:
003424                                  MOV     #5,R0           ;;LOAD R0 WITH TEST NUMBER
003424  012700  000005          1$:     CLR     @#PSW           ;CLEAR THE PSW
003430  005037  177776                  CCC                     ;SCOPE SYNC
003434  000257
                                2$:
003436                                  BPL     TST6            ;;TEST THE BPL - IT SHOULD BR
003436  100002
                                3$:     HALT                    ;BPL FAILED TO LOAD THE PC
003440  000000                          BR      1$              ;LOCK ON HARD ERROR
003442  000772
                                ;;**************************************************************
                                ;*TEST 6        BASIC "MOV  (R4),R8" TEST - (R4)=177776
                                ;;**************************************************************
                                TST6:
003444                                  MOV     #6,R0           ;;LOAD R0 WITH TEST NUMBER
003444  012700  000006                  MOV     #PSW,R5         ;SOURCE ADDR = 177776
003450  012705  177776          1$:     CLR     (R5)            ;MAKE [PSW]=000
003454  005015                          CLR     R3              ;[DEST] = 000000
003456  005003                          SCC                     ;MAKE [PSW]=017
003460  000277
                                2$:     MOV     (R5),R3         ;TEST THE MOV
003462  011503
                                        CMP     R3,#17          ;CORRECT RESULT ?
003464  020327  000017                  BEQ     TST7            ;;BR IF YES
003470  001402
```

```
2039
2040  003472  000000                  3$:     HALT                    ;ERROR-MOV FAILED
2041  003474  000767                          BR      1$              ;LOCK ON HARD ERROR
2042                                  ;;**********************************************************************
2043                                  ;*TEST 7       BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
2044                                  ;;**********************************************************************
2045  003476                          TST7:
2046  003476  012700  000007                 MOV     #7,R0           ;;LOAD R0 WITH TEST NUMBER
2047  003502  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2048  003506  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2049  003512  012737  125252  063312 1$:     MOV     #125252,@#MBUF0 ;MAKE [DEST] = 125252
2050  003520  000257                         CCC                     ;MAKE N:C=0000
2051
2052  003522  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2053
2054  003524  001402                          BEQ     TST10           ;;BR IF "Z" GOT SET
2055
2056  003526  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO SET "Z"
2057  003530  000770                          BR      1$              ;LOCK ON HARD ERROR
2058                                  ;;**********************************************************************
2059                                  ;*TEST 10      BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
2060                                  ;;**********************************************************************
2061  003532                          TST10:
2062  003532  012700  000010                 MOV     #10,R0          ;;LOAD R0 WITH TEST NUMBER
2063  003536  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2064  003542  012704  000001                 MOV     #1,R4           ;RESULT S / B = 000001
2065  003546  005037  063312          1$:    CLR     @#MBUF0         ;MAKE [DEST] = 000000
2066  003552  000264                         SEZ                     ;MAKE N:C=0100
2067
2068  003554  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2069
2070  003556  001002                          BNE     TST11           ;;BR IF "Z" GOT CLEARED
2071
2072  003560  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO CLR "Z"
2073  003562  000771                          BR      1$              ;LOCK ON HARD ERROR
2074
2075                                  ;;**********************************************************************
2076                                  ;*TEST 11      BASIC "CMP #N,R" TEST - N = [R]
2077                                  ;;**********************************************************************
2078  003564                          TST11:
2079  003564  012700  000011                 MOV     #11,R0          ;;LOAD R0 WITH TEST NUMBER
2080  003570  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2081  003574  010403                  1$:    MOV     R4,R3           ;[DEST] = 125252
2082  003576  000257                         CCC                     ;SCOPE SYNC
2083
2084  003600  022703  125252          2$:    CMP     #125252,R3      ;TEST THE CMP
2085
2086  003604  001402                          BEQ     4$              ;BR IF N = [R]
2087
2088  003606  000000                  3$:     HALT                    ;CMP FAILED
2089  003610  000771                          BR      1$              ;LOCK ON HARD ERROR
2090
2091  003612  020403                  4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
2092  003614  001402                          BEQ     TST12           ;;BR IF NO
2093
2094  003616  000000                  5$:     HALT                    ;CMP DELIVERED A RESULT
```

```
2095  003620  000765                          BR      1$              ;LOCK ON HARD ERROR
2096
2097                                  ;;*****************************************************************
2098                                  ;*TEST 12       BASIC "CMP #N,R" TEST - N NOT EQUAL TO [R]
2099                                  ;;*****************************************************************
2100  003622                          TST12:
2101  003622  012700  000012                 MOV     #12,R0          ;;LOAD R0 WITH TEST NUMBER
2102  003626  005004                         CLR     R4              ;RESULT S / B = 000000
2103  003630  010403                  1$:    MOV     R4,R3           ;[DEST] = 125252
2104  003632  000264                         SEZ                     ;SCOPE SYNC
2105
2106  003634  022703  000001          2$:    CMP     #1,R3           ;TEST THE CMP
2107
2108  003640  001002                         BNE     4$              ;BR IF N NOT EQUAL TO [R]
2109
2110  003642  000000                  3$:    HALT                    ;CMP FAILED
2111  003644  000771                         BR      1$              ;LOCK ON HARD ERROR
2112
2113  003646  020403                  4$:    CMP     R4,R3           ;DID CMP ALTER [DEST]?
2114  003650  001402                         BEQ     TST13           ;;BR IF NO
2115
2116  003652  000000                  5$:    HALT                    ;CMP DELIVERED A RESULT
2117  003654  000765                         BR      1$              ;LOCK ON HARD ERROR
2118
2119                                  ;;*****************************************************************
2120                                  ;*TEST 13       BASIC "MOV  RA,(RB)" TEST
2121                                  ;;*****************************************************************
2122  003656                          TST13:
2123  003656  012700  000013                 MOV     #13,R0          ;;LOAD R0 WITH TEST NUMBER
2124  003662  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR=MBUF0
2125  003666  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
2126  003672  005012                  1$:    CLR     (R2)            ;MAKE [DEST] = 000000
2127  003674  000257                         CCC                     ;SCOPE SYNC - N:C=0000
2128
2129  003676  010412                  2$:    MOV     R4,(R2)         ;TEST THE MOV
2130
2131  003700  020412                         CMP     R4,(R2)         ;RESULT CORRECT ?
2132  003702  001402                         BEQ     TST14           ;;BR IF  YES
2133
2134  003704  000000                  3$:    HALT                    ;ERROR - MOV FAILED
2135  003706  000771                         BR      1$              ;LOCK ON HARD ERROR
2136
2137                                  ;;*****************************************************************
2138                                  ;*TEST 14       BASIC "MOV  #N,(R)" TEST
2139                                  ;;*****************************************************************
2140  003710                          TST14:
2141  003710  012700  000014                 MOV     #14,R0          ;;LOAD R0 WITH TEST NUMBER
2142  003714  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2143  003720  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
2144  003724  005012                  1$:    CLR     (R2)            ;MAKE [DEST] = 000000
2145  003726  000257                         CCC                     ;SCOPE SYNC
2146
2147  003730  012712  177777          2$:    MOV     #-1,(R2)        ;TEST THE MOV
2148
2149  003734  020412                         CMP     R4,(R2)         ;RESULT OK ?
2150  003736  001402                         BEQ     TST15           ;;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 15       BASIC "MOVB #N,X(R)" TEST - DEST EVEN
                                ;;*****************************************************************
003744                          TST15:
003744  012700  000015          	MOV     #15,R0          ;;LOAD R0 WITH TEST NUMBER
003750  012704  177401          	MOV     #177401,R4      ;RESULT S / B = 177401
003754  012702  063316          	MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
003760  012705  063312          	MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
003764  012712  177777          1$:     MOV     #-1,(R2)        ;[DEST] = 177777
003770  000257                  	CCC                     ;SCOPE SYNC

003772  112765  000001  000004  2$:     MOVB    #1,4(R5)        ;TEST THE MOVB

004000  020412                  	CMP     R4,(R2)         ;RESULT OK?
004002  001402                  	BEQ     TST16           ;;BR IF YES

004004  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
004006  000766                  	BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 16       BASIC "MOVB #N,X(R)" TEST - DEST ODD
                                ;;*****************************************************************
004010                          TST16:
004010  012700  000016          	MOV     #16,R0          ;;LOAD R0 WITH TEST NUMBER
004014  012704  000777          	MOV     #777,R4         ;RESULT S / B = 777
004020  012702  063316          	MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
004024  012705  063312          	MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
004030  012712  177777          1$:     MOV     #-1,(R2)        ;[DEST] = 177777
004034  000257                  	CCC                     ;SCOPE SYNC

004036  112765  000001  000005  2$:     MOVB    #1,5(R5)        ;TEST THE MOVB

004044  020412                  	CMP     R4,(R2)         ;RESULT OK?
004046  001402                  	BEQ     TST17           ;;BR IF YES

004050  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
004052  000766                  	BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 17       BASIC "TST  @#A" TEST WITH [A] GT 0
                                ;;*****************************************************************
004054                          TST17:
004054  012700  000017          	MOV     #17,R0          ;;LOAD R0 WITH TEST NUMBER
004060  012702  063312          	MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004064  012704  000377          	MOV     #377,R4         ;RESULT S / B = 377 (NO CHANGE)
004070  010412                  1$:     MOV     R4,(R2)         ;[DEST] = 377
004072  000257                  	CCC                     ;SCOPE SYNC

004074  005737  063312          2$:     TST     @#MBUF0         ;TEST THE TST

004100  001401                  	BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
004102  100002                  	BPL     TST20           ;;BR IF "N" CLEAR - IT SHOULD BE
```

```
2207
2208  004104  000000                  3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
2209  004106  000770                          BR      1$              ;LOCK ON HARD ERROR
2210                                  ;;*****************************************************************
2211                                  ;*TEST 20       BASIC "TST @#A" TEST WITH (A) LT 0
2212                                  ;;*****************************************************************
2213  004110                          TST20:
2214  004110  012700  000020                 MOV     #20,R0          ;;LOAD R0 WITH TEST NUMBER
2215  004114  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2216  004120  012704  100000                 MOV     #100000,R4      ;MAKE S / B = 100000
2217  004124  010412                  1$:    MOV     R4,(R2)         ;MAKE (DEST) = 100000
2218  004126  000257                         CCC                     ;SCOPE SYNC
2219
2220  004130  005737  063312          2$:    TST     @#MBUF0         ;TEST THE TST
2221
2222  004134  001401                         BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
2223  004136  100402                         BMI     4$              ;BR IF "N" SET - IT SHOULD BE
2224
2225  004140  000000                  3$:    HALT                    ;TST FAILED TO ALTER CODES PROPERLY
2226  004142  000770                         BR      1$              ;LOCK ON HARD ERROR
2227  004144  020412                  4$:    CMP     R4,(R2)         ;DID TST DISTURB (DEST) ?
2228  004146  001402                         BEQ     TST21           ;;BR IF NOT
2229
2230  004150  000000                  5$:    HALT                    ;TST DELIVERED A RESULT
2231  004152  000764                         BR      1$              ;LOCK ON HARD ERROR
2232
2233                                  ;;*****************************************************************
2234                                  ;*TEST 21       BASIC "TST @#A" WITH (A) = 0
2235                                  ;;*****************************************************************
2236  004154                          TST21:
2237  004154  012700  000021                 MOV     #21,R0          ;;LOAD R0 WITH TEST NUMBER
2238  004160  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2239  004164  005004                         CLR     R4              ;RESULT S / B = 0 (IT SHOULDN'T CHANGE
2240  004166  005012                  1$:    CLR     (R2)            ;(DEST) = 0
2241  004170  000257                         CCC                     ;SCOPE SYNC - Z=0
2242
2243  004172  005737  063312          2$:    TST     @#MBUF0         ;TEST THE TST
2244
2245  004176  001402                         BEQ     4$              ;BR IF TST SET "Z"
2246
2247  004200  000000                  3$:    HALT                    ;TST FAILED TO SET "Z"
2248  004202  000771                         BR      1$              ;LOCK ON HARD ERROR
2249
2250  004204  020412                  4$:    CMP     R4,(R2)         ;(DEST) STILL = 000000
2251  004206  001402                         BEQ     TST22           ;;BR IF YES
2252
2253  004210  000000                  5$:    HALT                    ;TST ALTERED THE (DEST)
2254  004212  000765                         BR      1$              ;LOCK ON HARD ERROR
2255
2256                                  ;;*****************************************************************
2257                                  ;*TEST 22       BASIC "BIT #N,@#A" WITH BIT SET IN "A"
2258                                  ;;*****************************************************************
2259  004214                          TST22:
2260  004214  012700  000022                 MOV     #22,R0          ;;LOAD R0 WITH TEST NUMBER
2261  004220  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2262  004224  012704  040000                 MOV     #40000,R4       ;RESULT S / B = 40000
```

```
2263  004230  010412                  1$:     MOV     R4,(R2)         ;MAKE [DEST] = 40000
2264  004232  000277                          SCC                     ;SCOPE SYNC - Z=1
2265
2266  004234  032737  040000  063312  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
2267
2268  004242  001002                          BNE     TST23           ;;BR IF Z=0 - IT SHOULD BE
2269
2270  004244  000000                  3$:     HALT                    ;BIT FAILED TO CLEAR "Z"
2271  004246  000770                          BR      1$              ;LOCK ON HARD ERROR
2272
2273                                  ;;**********************************************************************
2274                                  ;*TEST 23       BASIC "BIT  #N,@#A" WITH BIT CLEAR IN "A"
2275                                  ;;**********************************************************************
2276  004250                          TST23:
2277  004250  012700  000023                  MOV     #23,R0          ;;LOAD R0 WITH TEST NUMBER
2278  004254  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2279  004260  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2280  004262  000257                          CCC                     ;SCOPE SYNC - Z=0
2281
2282  004264  032737  040000  063312  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
2283
2284  004272  001402                          BEQ     4$              ;BR IF Z=1 - IT SHOULD BE
2285
2286  004274  000000                  3$:     HALT                    ;BIT FAILED TO SET "Z"
2287  004276  000770                          BR      1$              ;LOCK ON HARD ERROR
2288
2289  004300  005712                  4$:     TST     (R2)            ;DID BIT DELIVER A RESULT
2290  004302  001402                          BEQ     TST24           ;;BR IF NOT
2291
2292  004304  000000                  5$:     HALT                    ;BIT DISTURBED THE [DEST]
2293  004306  000764                          BR      1$              ;LOCK ON HARD ERROR
2294
2295                                  ;;**********************************************************************
2296                                  ;*TEST 24       BASIC "TST  (R)+" TEST
2297                                  ;;**********************************************************************
2298  004310                          TST24:
2299  004310  012700  000024                  MOV     #24,R0          ;;LOAD R0 WITH TEST NUMBER
2300                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT0
2301  004314  032737  000001  063234          BIT     #BIT0,@#BPTLOC  ;BREAKPOINT HALT SET ??
2302  004322  001401                          BEQ     .+4             ;BR IF NOT
2303  004324  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2304  004326  012702  063312                  MOV     #MBUF0,R2       ;INITIAL DEST ADDR = MBUF0
2305  004332  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2306  004334  000257                          CCC                     ;SCOPE SYNC
2307
2308  004336  005722                  2$:     TST     (R2)+           ;TEST THE TST
2309
2310  004340  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2311
2312  004342  000000                  3$:     HALT                    ;TST FAILED TO SET "Z"
2313  004344  000772                          BR      1$              ;LOCK ON HARD ERROR
2314
2315  004346  022702  063314          4$:     CMP     #MBUF0+2,R2     ;DID REG. GET AUTO-INCREMENTED ?
2316  004352  001402                          BEQ     TST25           ;;BR IF YES
2317
2318  004354  000000                  5$:     HALT                    ;TST FAILED TO UPDATE REGISTER
```

```
2319  004356  000765                    BR      1$              ;LOCK ON HARD ERROR
2320
2321                                    ;;*****************************************************************
2322                                    ;*TEST 25        BASIC "TST -(R)" TEST
2323                                    ;;*****************************************************************
2324  004360                            TST25:
2325  004360  012700  000025            MOV     #25,R0          ;;LOAD R0 WITH TEST NUMBER
2326  004364  012702  063330            MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2327  004370  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
2328  004374  012705  063332  1$:       MOV     #DWTA+10,R5     ;BASE DEST ADDR = DWTA+10
2329  004400  000270                    SEN                     ;SCOPE SYNC
2330
2331  004402  005745            2$:     TST     -(R5)           ;TEST THE TST
2332
2333  004404  100002                    BPL     4$              ;BR IF "N" CLEAR
2334
2335  004406  000000            3$:     HALT                    ;TST FAILED TO CLEAR "N"
2336  004410  000771                    BR      1$              ;LOCK ON HARD ERROR
2337
2338  004412  020502            4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED?
2339  004414  001402                    BEQ     6$              ;BR IF YES
2340
2341  004416  000000            5$:     HALT                    ;ERROR - TST FAILED TO UPDATE DEST REG
2342  004420  000765                    BR      1$              ;LOCK ON HARD ERROR
2343
2344  004422  020412            6$:     CMP     R4,(R2)         ;DID TST ALTER [DEST]?
2345  004424  001403                    BEQ     TST26           ;;BR IF NOT
2346
2347  004426  000000            7$:     HALT                    ;TST ALTERED [DEST]
2348  004430  010412                    MOV     R4,(R2)         ;RESTORE [DEST]
2349  004432  000760                    BR      1$              ;LOCK ON HARD ERROR
2350
2351                                    ;;*****************************************************************
2352                                    ;*TEST 26        BASIC "COM  @#A" TEST
2353                                    ;;*****************************************************************
2354  004434                            TST26:
2355  004434  012700  000026            MOV     #26,R0          ;;LOAD R0 WITH TEST NUMBER
2356  004440  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2357  004444  005004                    CLR     R4              ;RESULT S / B = 177777
2358  004446  005104                    COM     R4
2359  004450  005012            1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2360  004452  000257                    CCC                     ;SCOPE SYNC
2361
2362  004454  005137  063312    2$:     COM     @#MBUF0 ;TEST THE COM
2363
2364  004460  020412                    CMP     R4,(R2)         ;RESULT = 177777 ??
2365  004462  001402                    BEQ     TST27           ;;BR IF YES
2366
2367  004464  000000            3$:     HALT                    ;COM DELIVERED THE WRONG RESULT
2368  004466  000770                    BR      1$
2369
2370                                    ;;*****************************************************************
2371                                    ;*TEST 27        BASIC "INC  @#A" TEST
2372                                    ;;*****************************************************************
2373  004470                            TST27:
2374  004470  012700  000027            MOV     #27,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2375  004474  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2376  004500  012704  000100          MOV     #100,R4         ;RESULT S / B = 100
2377  004504  012712  000077  1$:     MOV     #77,(R2)        ;[DEST] = 77
2378  004510  000257                  CCC                     ;SCOPE SYNC
2379
2380  004512  005237  063312  2$:     INC     @#MBUF0         ;TEST THE INC
2381
2382  004516  020412                  CMP     R4,(R2)         ;DID RESULT = 100 ??
2383  004520  001402                  BEQ     TST30           ;;BR IF YES
2384
2385  004522  000000  3$:     HALT                            ;INC DELIVERED WRONG RESULT
2386  004524  000767                  BR      1$              ;LOCK ON HARD ERROR
2387
2388                          ;;**************************************************************
2389                          ;*TEST 30        BASIC "DEC  RN" TEST
2390                          ;;**************************************************************
2391  004526                  TST30:
2392  004526  012700  000030          MOV     #30,R0          ;;LOAD R0 WITH TEST NUMBER
2393  004532  012703  000001  1$:     MOV     #1,R3           ;[DEST] = +1
2394  004536  000257                  CCC                     ;SCOPE SYNC
2395
2396  004540  005303  2$:     DEC     R3                      ;TEST THE DEC
2397
2398  004542  005703                  TST     R3              ;RESULT = 000000 ??
2399  004544  001402                  BEQ     TST31           ;;BR IF YES
2400
2401  004546  000000  3$:     HALT                            ;DEC DELIVERED THE WRONG RESULT
2402  004550  000770                  BR      1$              ;LOCK ON HARD ERROR
2403
2404                          ;;**************************************************************
2405                          ;*TEST 31        BASIC "DEC  @#A" TEST
2406                          ;;**************************************************************
2407  004552                  TST31:
2408  004552  012700  000031          MOV     #31,R0          ;;LOAD R0 WITH TEST NUMBER
2409  004556  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
2410  004562  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2411  004566  005012  1$:     CLR     (R2)                    ;MAKE [DEST] = 000000
2412  004570  000257                  CCC                     ;SCOPE SYNC
2413
2414  004572  005337  063312  2$:     DEC     @#MBUF0         ;TEST THE DEC
2415
2416  004576  020412                  CMP     R4,(R2)         ;DID RESULT = 177777 ??
2417  004600  001402                  BEQ     TST32           ;;BR IF YES
2418
2419  004602  000000  3$:     HALT                            ;DEC DELIVERED WRONG RESULT
2420  004604  000770                  BR      1$              ;LOCK ON HARD ERROR
2421
2422                          ;;**************************************************************
2423                          ;*TEST 32        BASIC "CLR X(R)" TESTS
2424                          ;;**************************************************************
2425  004606                  TST32:
2426  004606  012700  000032          MOV     #32,R0          ;;LOAD R0 WITH TEST NUMBER
2427  004612  012702  063314          MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
2428  004616  005004                  CLR     R4              ;RESULT S / B = 000000
2429  004620  012705  063312  1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2430  004624  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
```

```
2431  004630  000257                    CCC                     ;SCOPE SYNC
2432
2433  004632  005065  000002  2$:       CLR     2(R5)           ;TEST THE CLR
2434
2435  004636  020412                    CMP     R4,(R2)         ;RESULT = 0?
2436  004640  001402                    BEQ     TST33           ;;BR IF YES
2437
2438  004642  000000            3$:     HALT                    ;CLR FAILED TO ZERO [DEST]
2439  004644  000765                    BR      1$              ;LOCK ON HARD ERROR.
2440
2441                            ;;**********************************************************************
2442                            ;*TEST 33       BASIC "ASL  RN" TEST WITH [DEST]=125252 AND C(0)
2443                            ;;**********************************************************************
2444  004646                    TST33:
2445  004646  012700  000033            MOV     #33,R0          ;;LOAD R0 WITH TEST NUMBER
2446  004652  012703  125252    1$:     MOV     #125252,R3      ;MAKE [DEST] = 125252
2447  004656  000257                    CCC                     ;MAKE C=0
2448
2449  004660  006303            2$:     ASL     R3              ;TEST THE ASL - IT SHOULD SET "C"
2450
2451  004662  103402                    BCS     4$              ;BR IF "C" GOT SET
2452
2453  004664  000000            3$:     HALT                    ;ASL FAILED TO SET "C" BIT
2454  004666  000771                    BR      1$              ;LOCK ON HRD ERROR
2455
2456  004670  022703  052524    4$:     CMP     #52524,R3       ;WAS RESULT = 52524 ??
2457  004674  001402                    BEQ     TST34           ;;BR IF YES
2458
2459  004676  000000            5$:     HALT                    ;ASL DELIVERED THE WRONG RESULT
2460  004700  000764                    BR      1$              ;LOCK ON HARD ERROR
2461
2462                            ;;**********************************************************************
2463                            ;*TEST 34       BASIC "ASL  RN" TEST WITH [DEST]=052525 AND C(1)
2464                            ;;**********************************************************************
2465  004702                    TST34:
2466  004702  012700  000034            MOV     #34,R0          ;;LOAD R0 WITH TEST NUMBER
2467  004706  012703  052525    1$:     MOV     #052525,R3      ;MAKE [DEST] = 052525
2468  004712  000261                    SEC                     ;MAKE C=1
2469
2470  004714  006303            2$:     ASL     R3              ;TEST THE ASL - IT SHOULD CLR "C"
2471
2472  004716  103002                    BCC     4$              ;BR IF "C" GOT CLEARED
2473
2474  004720  000000            3$:     HALT                    ;ASL FAILED TO CLEAR "C"
2475  004722  000771                    BR      1$              ;LOCK ON HARD ERROR
2476
2477  004724  022703  125252    4$:     CMP     #125252,R3      ;RESULT = 125252 ??
2478  004730  001402                    BEQ     TST35           ;;BR IF YES
2479
2480  004732  000000            5$:     HALT                    ;ASL DELIVERED WRONG REULT
2481  004734  000764                    BR      1$              ;LOCK ON HARD ERROR
2482                            ;;**********************************************************************
2483                            ;*TEST 35       BASIC "ROL  RN" TEST WITH [DEST]=125252 AND C(0)
2484                            ;;**********************************************************************
2485  004736                    TST35:
2486  004736  012700  000035            MOV     #35,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2487  004742  012703  125252          1$:     MOV     #125252,R3      ;MAKE [DEST] = 125252
2488  004746  000257                          CCC                     ;MAKE C=0
2489
2490  004750  006103                  2$:     ROL     R3              ;TEST THE ROL - IT SHOULD SET C
2491
2492  004752  103402                          BCS     4$              ;BR IF "C" GOT SET
2493
2494  004754  000000                  3$:     HALT                    ;ROL FAILED TO SET "C"
2495  004756  000771                          BR      1$              ;LOCK ON HARD ERROR
2496
2497  004760  022703  052524          4$:     CMP     #052524,R3      ;RESULT = 052524 ??
2498  004764  001402                          BEQ     TST36           ;;BR IF YES
2499
2500  004766  000000                  5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2501  004770  000764                          BR      1$              ;LOCK ON HARD ERROR
2502
2503                                  ;;**********************************************************************
2504                                  ;*TEST 36       BASIC "ROL  RN" TEST WITH [DEST]=052524 AND C(1)
2505                                  ;;**********************************************************************
2506  004772                          TST36:
2507  004772  012700  000036                  MOV     #36,R0          ;;LOAD R0 WITH TEST NUMBER
2508  004776  012703  052524          1$:     MOV     #052524,R3      ;MAKE [DEST] = 052524
2509  005002  000261                          SEC                     ;MAKE C=1
2510
2511  005004  006103                  2$:     ROL     R3              ;TEST THE ROL - IT SHOULD CLEAR C
2512
2513  005006  103002                          BCC     4$              ;BR IF "C" IS CLEAR
2514
2515  005010  000000                  3$:     HALT                    ;ROL FAILED TO CLEAR "C"
2516  005012  000771                          BR      1$              ;LOCK ON HARD ERROR
2517
2518  005014  022703  125251          4$:     CMP     #125251,R3      ;RESULT = 125251 ??
2519  005020  001402                          BEQ     TST37           ;;BR IF YES
2520
2521  005022  000000                  5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2522  005024  000764                          BR      1$              ;LOCK ON HARD ERROR
2523
2524                                  ;;**********************************************************************
2525                                  ;*TEST 37       BASIS "TSTB (R)" TEST - EVEN ADDRESS
2526                                  ;;**********************************************************************
2527  005026                          TST37:
2528  005026  012700  000037                  MOV     #37,R0          ;;LOAD R0 WITH TEST NUMBER
2529  005032  012702  063330                  MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2530  005036  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
2531  005042  000257                  1$:     CCC                     ;SCOPE SYNC
2532
2533  005044  105712                  2$:     TSTB    (R2)            ;TEST THE TSTB
2534
2535  005046  100402                          BMI     4$              ;BR IF "N" SET - IT SHOULD BE
2536
2537  005050  000000                  3$:     HALT                    ;TSTB FAILED TO SET "N"
2538  005052  000773                          BR      1$              ;LOCK ON HARD ERROR
2539
2540  005054  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2541  005056  001403                          BEQ     TST40           ;;BR IF NOT
2542
```

```
2543  005060  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
2544  005062  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2545  005064  000766                          BR      1$              ;LOCK ON HARD EROR
2546                                  ;;**********************************************************************
2547                                  ;*TEST 40        BASIS "TSTB (R)" TEST - ODD ADDRESS
2548                                  ;;**********************************************************************
2549  005066                          TST40:
2550  005066  012700  000040                 MOV     #40,R0          ;;LOAD R0 WITH TEST NUMBER
2551  005072  012702  064040                 MOV     #DWTB+6,R2      ;DEST ADDR = DWTB+6
2552  005076  012704  177401                 MOV     #177401,R4      ;RESULT S / B = 177401
2553  005102  012703  064041                 MOV     #DWTB+7,R3      ;DEST ADDR USED = DWTB+7
2554  005106  000257                  1$:     CCC                     ;SCOPE SYNC
2555
2556  005110  105713                  2$:     TSTB    (R3)            ;TEST THE TSTB
2557
2558  005112  100402                          BMI     4$              ;BR IF "N" SET - IT SHOULD BE
2559
2560  005114  000000                  3$:     HALT                    ;TSTB FAILED TO SET "N"
2561  005116  000773                          BR      1$              ;LOCK ON HARD ERROR
2562
2563  005120  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2564  005122  001403                          BEQ     TST41           ;;BR IF NOT
2565
2566  005124  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
2567  005126  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2568  005130  000766                          BR      1$              ;LOCK ON HARD EROR
2569
2570                                  ;;**********************************************************************
2571                                  ;*TEST 41        BASIC "TSTB @#A" TEST - EVEN ADDRESS
2572                                  ;;**********************************************************************
2573  005132                          TST41:
2574  005132  012700  000041                 MOV     #41,R0          ;;LOAD R0 WITH TEST NUMBER
2575  005136  012702  063326                 MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
2576  005142  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 177400
2577  005146  000257                  1$:     CCC                     ;SCOPE SYNC
2578
2579  005150  105737  063326          2$:     TSTB    @#DWTA+4        ;TEST THE TSTB
2580
2581  005154  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2582
2583  005156  000000                  3$:     HALT                    ;TSTB FAILED TO SET "Z"
2584  005160  000772                          BR      1$              ;LOCK ON HARD ERROR
2585
2586  005162  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
2587  005164  001403                          BEQ     TST42           ;;BR IF NOT
2588
2589  005166  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
2590  005170  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2591  005172  000765                          BR      1$              ;LOCK ON HARD ERROR
2592
2593                                  ;;**********************************************************************
2594                                  ;*TEST 42        BASIC "TSTB @#A" TEST - ODD ADDRESS
2595                                  ;;**********************************************************************
2596  005174                          TST42:
2597  005174  012700  000042                 MOV     #42,R0          ;;LOAD R0 WITH TEST NUMBER
2598  005200  012702  063330                 MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
```

```
2599  005204  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
2600  005210  000257          1$:     CCC                     ;SCOPE SYNC
2601
2602  005212  105737  063331  2$:     TSTB    @#DWTA+7        ;TEST THE TSTB
2603
2604  005216  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2605
2606  005220  000000          3$:     HALT                    ;TSTB FAILED TO SET "Z"
2607  005222  000772                  BR      1$              ;LOCK ON HARD ERROR
2608
2609  005224  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
2610  005226  001403                  BEQ     TST43           ;;BR IF NOT
2611
2612  005230  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2613  005232  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2614  005234  000765                  BR      1$              ;LOCK ON HARD ERROR
2615
2616                          ;;**********************************************************************
2617                          ;*TEST 43        BASIC "DECB 1(SP)"
2618                          ;;**********************************************************************
2619  005236                  TST43:
2620  005236  012700  000043          MOV     #43,R0          ;;LOAD R0 WITH TEST NUMBER
2621  005242  010605                  MOV     SP,R5           ;SAVE SP
2622  005244  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
2623  005250  010506          1$:     MOV     R5,SP
2624  005252  005046                  CLR     -(SP)           ;[DEST] = 000000
2625  005254  000257                  CCC                     ;SCOPE SYNC
2626
2627  005256  105366  000001  2$:     DECB    1(SP)           ;TEST THE DECB
2628
2629  005262  020416                  CMP     R4,(SP)         ;RESULT = 177400?
2630  005264  001402                  BEQ     4$              ;BR IF YES
2631
2632  005266  000000          3$:     HALT                    ;ERROR - DECB FAILED
2633  005270  000767                  BR      1$              ;LOCK ON HARD ERROR
2634
2635  005272  010506          4$:     MOV     R5,SP           ;RESET THE SP
2636
2637                          ;;**********************************************************************
2638                          ;*TEST 44        BASIC "MOV @#A,R" TEST
2639                          ;;**********************************************************************
2640  005274                  TST44:
2641  005274  012700  000044          MOV     #44,R0          ;;LOAD R0 WITH TEST NUMBER
2642  005300  005003          1$:     CLR     R3              ;[DEST] = 000000
2643  005302  000257                  CCC                     ;SCOPE SYNC
2644
2645  005304  013703  063276  2$:     MOV     @#ATA,R3        ;TEST THE MOV
2646
2647  005310  022703  063322          CMP     #DWTA,R3        ;RESULT = DWTA?
2648  005314  001402                  BEQ     TST45           ;;BR IF YES
2649
2650  005316  000000          3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2651  005320  000767                  BR      1$              ;LOCK ON HARD ERROR
2652
2653                          ;;**********************************************************************
2654                          ;*TEST 45        BASIC "MOV #N,X(R)" TEST
```

```
                                ;;*****************************************************************
005322                          TST45:
005322 012700 000045            MOV     #45,R0          ;;LOAD R0 WITH TEST NUMBER
005326 012702 063314            MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
005332 012704 125252            MOV     #125252,R4      ;RESULT S / B = 125252
005336 012703 063312    1$:     MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
005342 005012                   CLR     (R2)            ;[DEST] = 000000
005344 000257                   CCC                     ;SCOPE SYNC

005346 012763 125252 000002 2$: MOV     #125252,2(R3)   ;TEST THE MOV

005354 020412                   CMP     R4,(R2)         ;RESULT OK?
005356 001402                   BEQ     TST46           ;;BR IF YES

005360 000000           3$:     HALT                    ;MOV DELIVERED WRONG RESULT
005362 000765                   BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 46        BASIC "MOV #N,(R)" TEST
                                ;;*****************************************************************
005364                          TST46:
005364 012700 000046            MOV     #46,R0          ;;LOAD R0 WITH TEST NUMBER
005370 012703 063312            MOV     #MBUF0,R3       ;DEST ADDR = MBUF0
005374 012704 125252            MOV     #125252,R4      ;RESULT S / B = 125252
005400 005013           1$:     CLR     (R3)            ;[DEST] = 000000
005402 000257                   CCC                     ;SCOPE SYNC

005404 012713 125252    2$:     MOV     #125252,(R3)    ;TEST THE MOV

005410 020413                   CMP     R4,(R3)         ;RESULT OK?
005412 001402                   BEQ     TST47           ;;BR IF YES

005414 000000           3$:     HALT                    ;MOV DELIVERED WRONG RESULT
005416 000770                   BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 47        BASIC "MOV (RA)+,RB" TEST
                                ;;*****************************************************************
005420                          TST47:
005420 012700 000047            MOV     #47,R0          ;;LOAD R0 WITH TEST NUMBER
005424 012705 063276    1$:     MOV     #ATA,R5         ;SRC ADDR = ATA
005430 005003                   CLR     R3              ;[DEST] = 000000
005432 000257                   CCC                     ;SCOPE SYNC

005434 012503           2$:     MOV     (R5)+,R3        ;TEST THE MOV

005436 022703 063322            CMP     #DWTA,R3        ;RESULT OK?
005442 000402                   BR      4$              ;BR IF YES

005444 000000           3$:     HALT                    ;MOV DELIVERED WRONG RESULT
005446 000766                   BR      1$              ;LOCK ON HARD ERROR

005450 022705 063300    4$:     CMP     #ATA+2,R5       ;DID SRC REG GET INCREMENTED?
005454 001402                   BEQ     TST50           ;;BR IF YES

005456 000000           5$:     HALT                    ;MOV FAILED TO UPDATE SRC. REG.
```

```
2711  005460  000761                          BR      1$              ;LOCK ON HARD ERROR
2712
2713                                  ;;*****************************************************************
2714                                  ;*TEST 50       BASIC "MOV  @#A,@#B"
2715                                  ;;*****************************************************************
2716  005462                          TST50:
2717  005462  012700  000050                  MOV     #50,R0          ;;LOAD R0 WITH TEST NUMBER
2718  005466  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2719  005472  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2720  005476  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2721  005500  000257                          CCC                     ;SCOPE SYNC
2722
2723  005502  013737  063276  063316  2$:     MOV     @#ATA,@#MBUF1   ;TEST THE MOV
2724  005510  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ?
2725  005512  001402                          BEQ     TST51           ;;BR IF YES
2726
2727  005514  000000                  3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2728  005516  000767                          BR      1$              ;LOCK ON HARD ERROR
2729
2730                                  ;;*****************************************************************
2731                                  ;*TEST 51       BASIC "MOV   X(R),PC" TEST
2732                                  ;;*****************************************************************
2733  005520                          TST51:
2734  005520  012700  000051                  MOV     #51,R0          ;;LOAD R0 WITH TEST NUMBER
2735  005524  012705  005532          1$:     MOV     #2$,R5          ;[R5] = 2$ (BASE ADDRESS)
2736  005530  000257                          CCC                     ;SCOPE SYNC
2737
2738  005532  016507  000010          2$:     MOV     4$-2$(R5),PC    ;TEST THE MOV - GO TO NEXT TEST VIA 4$
2739
2740  005536  000000                  3$:     HALT                    ;MOV FAILED TO LOAD THE PC
2741  005540  000771                          BR      1$              ;LOCK ON HARD ERROR
2742
2743  005542  005544                  4$:     .+2                     ;POINTER TO NEXT TEST
2744
2745                                  ;;*****************************************************************
2746                                  ;*TEST 52       BASIC "MOV  @#A,(R)" TEST
2747                                  ;;*****************************************************************
2748  005544                          TST52:
2749  005544  012700  000052                  MOV     #52,R0          ;;LOAD R0 WITH TEST NUMBER
2750  005550  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2751  005554  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2752  005560  005012                  1$:     CLR     (R2)            ;MAKE [DEST]=000000
2753  005562  000257                          CCC                     ;SCOPE SYNC - Z=0
2754
2755  005564  013712  063276          2$:     MOV     @#ATA,(R2)      ;TEST THE MOV
2756
2757  005570  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ??
2758  005572  001402                          BEQ     TST53           ;;BR IF YES
2759
2760  005574  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2761  005576  000770                          BR      1$              ;LOCK ON HARD ERROR
2762
2763
2764                                  ;;*****************************************************************
2765                                  ;*TEST 53       BASIC "MOV  X(RA),RB" TEST
2766                                  ;;*****************************************************************
2766  005600                          TST53:
```

```
2767  005600  012700  000053          MOV     #53,R0          ;;LOAD R0 WITH TEST NUMBER
2768  005604  012705  063276          MOV     #ATA,R5         ;[R5] = BASE ADDR FOR SOURCE (ATA)
2769  005610  005003          1$:     CLR     R3              ;MAKE [DEST] = 000000
2770  005612  000257                  CCC                     ;SCOPE SYNC
2771
2772  005614  016503  000004  2$:     MOV     4(R5),R3        ;TEST THE MOV
2773
2774  005620  022703  064630          CMP     #OBTA,R3        ;RESULT = #OBTA ??
2775  005624  001402                  BEQ     TST54           ;;BR IF YES
2776
2777  005626  000000  3$:     HALT                            ;MOV DELIVERED WRONG RESULT
2778  005630  000767          BR      1$                      ;LOCK ON HARD ERROR
2779
2780                  ;;*****************************************************************
2781                  ;*TEST 54       BASIC "MOV  RA,-(RB)" TEST
2782                  ;;*****************************************************************
2783  005632          TST54:
2784  005632  012700  000054          MOV     #54,R0          ;;LOAD R0 WITH TEST NUMBER
2785  005636  012702  063312          MOV     #MBUF0,R2       ;FINAL DEST ADDR = MBUF0
2786  005642  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2787  005646  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = TEMP2 + 2
2788  005652  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2789  005654  000257                  CCC                     ;SCOPE SYNC
2790
2791  005656  010445  2$:     MOV     R4,-(R5)        ;TEST THE MOV
2792
2793  005660  020412          CMP     R4,(R2)         ;RESULT = 125252
2794  005662  001402          BEQ     4$              ;BR IF YES
2795
2796  005664  000000  3$:     HALT                    ;MOV DELIVERED THE WRONG ESULT
2797  005666  000767          BR      1$              ;LOCK ON HARD ERROR
2798
2799  005670  020205  4$:     CMP     R2,R5           ;DID REGISTER GET DECREMENTED ?
2800  005672  001402          BEQ     TST55           ;;BR IF YES
2801
2802  005674  000000  5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2803  005676  000763          BR      1$              ;LOCK ON HARD ERROR
2804
2805                  ;;*****************************************************************
2806                  ;*TEST 55       BASIC "MOV   @#A,-(R)" TEST
2807                  ;;*****************************************************************
2808  005700          TST55:
2809  005700  012700  000055          MOV     #55,R0          ;;LOAD R0 WITH TEST NUMBER
2810  005704  012704  063322          MOV     #OWTA,R4        ;RESULT S / B = #OWTA
2811  005710  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2812  005714  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = MBUF0+2
2813  005720  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2814  005722  000257                  CCC                     ;SCOPE SYNC
2815
2816  005724  013745  063276  2$:     MOV     @#ATA,-(R5)     ;TEST THE MOV
2817
2818  005730  020412          CMP     R4,(R2)         ;RESULT = 000000
2819  005732  001402          BEQ     4$              ;BR IF YES
2820
2821  005734  000000  3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2822  005736  000766          BR      1$              ;LOCK ON HARD ERROR
```

```
                                ;;*****************************************************************
                                ;*TEST 56       BASIC "MOV  (R),@#A" TEST
                                ;;*****************************************************************
005750                          TST56:
005750  012700  000056          4$:     MOV     #56,R0          ;;LOAD R0 WITH TEST NUMBER
005754  012702  063312          1$:     MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
005760  012704  063322          2$:     MOV     #DWTA,R4        ;RESULT S / B = #DWTA
005764  012705  063276          3$:     MOV     #ATA,R5         ;SOURCE ADDR = ATA
005770  005012                  5$:     CLR     (R2)            ;MAKE [DEST] = 000000
005772  000257                          CCC                     ;SCOPE SYNC
005774  011537  063312          6$:     MOV     (R5),@#MBUF0    ;TEST THE MOV
006000  020412                          CMP     R4,(R2)         ;RESULT = #DWTA ??
006002  001402                          BEQ     TST57           ;;BR IF YES
006004  000000                  7$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006006  000770                          BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 57       BASIC "MOV  -(R),@#A" TEST
                                ;;*****************************************************************
006010                          TST57:
006010  012700  000057          4$:     MOV     #57,R0          ;;LOAD R0 WITH TEST NUMBER
006014  012702  063312          1$:     MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006020  012704  063322          2$:     MOV     #DWTA,R4        ;RESULT S / B = #DWTA
006024  012705  063300          3$:     MOV     #ATA+2,R5       ;INITIAL SOURCE ADDR = ATA+2
006030  005012                  5$:     CLR     (R2)            ;MAKE [DEST] = 000000
006032  000257                          CCC                     ;SCOPE SYNC
006034  014537  063312          6$:     MOV     -(R5),@#MBUF0   ;TEST THE MOV
006040  020412                          CMP     R4,(R2)         ;RESULT = #DWTA ?
006042  001402                          BEQ     4$              ;BR IF YES
006044  000000                  7$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006046  000766                          BR      1$              ;LOCK ON HARD ERROR
006050  022705  063276          4$:     CMP     #ATA,R5         ;DID THE SRC REG GET DECREMENTED ?
006054  001402                          BEQ     TST60           ;;BR IF YES
006056  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REG
006060  000761                          BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 60       BASIC "MOV  (R0),R8" TEST
                                ;;*****************************************************************
006062                          TST60:
006062  012700  000060          4$:     MOV     #60,R0          ;;LOAD R0 WITH TEST NUMBER
006066  012705  063276          1$:     MOV     #ATA,R5         ;INITIAL SOURCE ADDR = ATA
```

```
2879  006072  005003                        CLR     R3              ;MAKE [DEST] = 000000
2880  006074  000257                        CCC                     ;SCOPE SYNC
2881
2882  006076  012503                  2$:   MOV     (R5)+,R3        ;TEST THE MOV
2883
2884  006100  022703  063322                CMP     #DWTA,R3        ;RESULT = #DWTA ?
2885  006104  001402                        BEQ     4$              ;BR IF YES
2886
2887  006106  000000                  3$:   HALT                    ;MOV DELIVERED WRONG RESULT
2888  006110  000766                        BR      1$              ;LOCK ON HARD ERROR
2889
2890  006112  022705  063300          4$:   CMP     #ATA+2,R5       ;DID SOURCE REG GET INCREMENTED
2891  006116  001402                        BEQ     TST61           ;;BR IF YES
2892
2893  006120  000000                  5$:   HALT                    ;MOV FAILED TO UPDATE SOURCE REGISTER
2894  006122  000761                        BR      1$              ;LOCK ON HARD ERROR
2895
2896                                  ;;*****************************************************************
2897                                  ;*TEST 61       BASIC "MOV   X(RA),RB" TEST
2898                                  ;;*****************************************************************
2899  006124                          TST61:
2900  006124  012700  000061                MOV     #61,R0          ;;LOAD R0 WITH TEST NUMBER
2901  006130  012705  063276                MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
2902  006134  005003                  1$:   CLR     R3              ;MAKE [DEST] = 000000
2903  006136  000257                        CCC                     ;SCOPE SYNC
2904
2905  006140  016503  000002          2$:   MOV     2(R5),R3        ;TEST THE MOV
2906
2907  006144  022703  064032                CMP     #DWTB,R3        ;RESULT = #DWTB ?
2908  006150  001402                        BEQ     TST62           ;;BR IF YES
2909
2910  006152  000000                  3$:   HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2911  006154  000767                        BR      1$              ;LOCK ON HARD ERROR
2912                                  ;;*****************************************************************
2913                                  ;*TEST 62       BASIC "MOV   @X(RA),RB" TEST
2914                                  ;;*****************************************************************
2915  006156                          TST62:
2916  006156  012700  000062                MOV     #62,R0          ;;LOAD R0 WITH TEST NUMBER
2917  006162  012737  063324  063314         MOV     #DWTA+2,@#MBUF0+2 ;SET UP ADDRESS TABLE MBUF0
2918  006170  012705  063312                MOV     #MBUF0,R5       ;BASE ADDRESS IN R5
2919  006174  005003                  1$:   CLR     R3              ;MAKE [DEST] = 000000
2920  006176  000257                        CCC                     ;SCOPE SYNC
2921
2922  006200  017503  000002          2$:   MOV     @2(R5),R3       ;TEST THE MOV
2923
2924  006204  022703  177777                CMP     #-1,R3          ;RESULT = 177777
2925  006210  001402                        BEQ     TST63           ;;BR IF YES
2926
2927  006212  000000                  3$:   HALT                    ;MOV DELIVERED THE WRONG RESULT
2928  006214  000767                        BR      1$              ;LOCK ON HARD ERROR
2929
2930                                  ;;*****************************************************************
2931                                  ;*TEST 63       BASIC "MOV  (R)+,X(R)" TEST
2932                                  ;;*****************************************************************
2933  006216                          TST63:
2934  006216  012700  000063                MOV     #63,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2935  006222  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2936  006226  012702  063320          MOV     #MBUF1+2,R2     ;FINAL DEST ADDR = MBUF1+2
2937  006232  010437  063312          MOV     R4,@#MBUF0      ;SOURCE OPERAND = 125252
2938  006236  012705  063312  1$:     MOV     #MBUF0,R5       ;[R5] = INITIAL SRC ADDR = MBUF0
2939  006242  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2940  006244  000257                  CCC                     ;SCOPE SYNC
2941
2942  006246  012565  000004  2$:     MOV     (R5)+,4(R5)     ;TEST THE MOV
2943
2944  006252  020412                  CMP     R4,(R2)         ;RESULT = 125252 ?
2945  006254  001402                  BEQ     4$              ;BR IF YES
2946
2947  006256  000000  3$:     HALT                            ;MOV    DELIVERED WRONG RESULT
2948  006260  000766                  BR      1$              ;LOCK ON HARD ERROR
2949
2950  006262  022705  063314  4$:     CMP     #MBUF0+2,R5     ;DID REGISTER GET INCREMENTED ?
2951  006266  001402                  BEQ     TST64           ;;BR IF YES
2952
2953  006270  000000  5$:     HALT                            ;MOV FAILED TO UPDATE REGISTER
2954  006272  000761                  BR      1$              ;LOCK ON HARD ERROR
2955
2956                          ;;*****************************************************************
2957                          ;*TEST 64       BASIC "CMP  R,@#A" TEST WITH [R] = [A]
2958                          ;;*****************************************************************
2959  006274                  TST64:
2960  006274  012700  000064          MOV     #64,R0          ;;LOAD R0 WITH TEST NUMBER
2961                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT1
2962  006300  032737  000002  063234  BIT     #BIT1,@#BPTLOC  ;BREAKPOINT HALT SET ??
2963  006306  001401                  BEQ     .+4             ;BR IF NOT
2964  006310  000000                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2965  006312  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2966  006316  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2967  006322  010405  1$:     MOV     R4,R5           ;[R5] = SOURCE OP = 125252
2968  006324  010412                  MOV     R4,(R2)         ;MAKE [DEST] = 125252
2969  006326  000257                  CCC                     ;SCOPE SYNC
2970
2971  006330  020537  063312  2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2972
2973  006334  001402                  BEQ     4$              ;BR IF "Z" WAS SET - IT SHOULD BE
2974
2975  006336  000000  3$:     HALT                            ;CMP FAILED TO SET "Z"
2976  006340  000770                  BR      1$              ;LOCK ON HARD ERROR
2977
2978  006342  020412  4$:     CMP     R4,(R2)         ;IS RESULT STILL = 125252 ?
2979  006344  001402                  BEQ     TST65           ;;BR IF YES
2980
2981  006346  000000  5$:     HALT                            ;CMP ALTERED [DEST]
2982  006350  000764                  BR      1$              ;LOCK ON HARD ERROR
2983
2984                          ;;*****************************************************************
2985                          ;*TEST 65       BASIC "CMP  R,@#A" WITH [R] NOT EQUAL TO [A]
2986                          ;;*****************************************************************
2987  006352                  TST65:
2988  006352  012700  000065          MOV     #65,R0          ;;LOAD R0 WITH TEST NUMBER
2989  006356  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2990  006362  012704  125252          MOV     #125252,R4      ;MAKE RESULT S / B = 125252
```

```
                                ;[R5] = SOURCE OP = 000000
                                ;MAKE [DEST] = 125252
                                ;SCOPE SYNC - MAKE Z=1

                                ;TEST THE CMP

                                ;;BR IF Z=0 - IT SHOULD BE

                                ;CMP FAILED TO CLEAR "Z"
                                ;LOCK ON HARD ERROR

                        ;;*****************************************************************
                        ;*TEST 66       BASIC "BIS #N,@#A" TEST - N=177777,[A]=000000
                        ;;*****************************************************************
                        TST66:
                                ;;LOAD R0 WITH TEST NUMBER
                                ;DEST ADDR = MBUF0
                                ;RESULT S / B = 177777
                                ;[DEST] = 000000
                                ;SCOPE SYNC

                                ;TEST THE BIS

                                ;RESULT OK?
                                ;;BR IF YES

                                ;BIS FAILED TO SET ALL BITS IN BITFLG
                                ;LOCK ON HARD ERROR

                        ;;*****************************************************************
                        ;*TEST 67       BASIC "BIC  #N,@#A" TEST
                        ;;*****************************************************************
                        TST67:
                                ;;LOAD R0 WITH TEST NUMBER
                                ;DEST ADDR = MBUF0
                                ;RESULT S / B = 77
                                ;MAKE [DEST] = 177777
                                ;SCOPE SYNC

                                ;TEST THE BIC

                                ;DID RESULT = 77 ?
                                ;;BR IF YES

                                ;BIC DELIVERED THE WRONG RESULT
                                ;LOCK ON HARD ERROR

                        ;;*****************************************************************
                        ;*TEST 70       BASIC "BIC #N,R" TEST
                        ;;*****************************************************************
                        TST70:
                                ;;LOAD R0 WITH TEST NUMBER
                                ;[DEST] = 177777

                                ;SCOPE SYNC
```

```
3047  006516  042703  177400          2$:     BIC     #177400,R3      ;TEST THE BIC
3048
3049  006522  022703  000377                  CMP     #377,R3         ;RESULT OK?
3050  006526  001402                          BEQ     TST71           ;;BR IF YES
3051
3052  006530  000000                  3$:     HALT                    ;BIC FAILED TO CLEAR HI-BYTE
3053  006532  000766                          BR      1$              ;LOCK ON HARD ERROR
3054
3055                                  ;;**********************************************************************
3056                                  ;*TEST 71        BASIC "BIC #N,2(SP)" TEST
3057                                  ;;**********************************************************************
3058  006534                          TST71:
3059  006534  012700  000071                  MOV     #71,R0          ;;LOAD R0 WITH TEST NUMBER
3060  006540  012704  000357                  MOV     #357,R4         ;RESULT S / B = 357
3061  006544  010605                          MOV     SP,R5           ;SAVE SP
3062  006546  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3063  006550  012746  000377                  MOV     #377,-(SP)      ;[DEST] = 377 PUT ON STACK
3064  006554  005746                          TST     -(SP)           ;DECREMENT SP
3065  006556  000257                          CCC                     ;SCOPE SYNC
3066
3067  006560  042766  000020  000002  2$:     BIC     #20,2(SP)       ;TEST THE BIC - CLEAR BIT 4
3068
3069  006566  010602                          MOV     SP,R2           ;[R2] = DEST ADDR
3070  006570  005722                          TST     (R2)+
3071  006572  020412                          CMP     R4,(R2)         ;RESULT = 357?
3072  006574  001402                          BEQ     4$              ;BR IF YES
3073
3074  006576  000000                  3$:     HALT                    ;BIC FAILED TO CLR BIT2 OF DEST
3075  006600  000762                          BR      1$              ;LOCK ON HARD ERROR
3076
3077  006602  010506                  4$:     MOV     R5,SP
3078
3079                                  ;;**********************************************************************
3080                                  ;*TEST 72        BASIC "ADD  #N,RN " TEST
3081                                  ;;**********************************************************************
3082  006604                          TST72:
3083  006604  012700  000072                  MOV     #72,R0          ;;LOAD R0 WITH TEST NUMBER
3084  006610  012703  000002          1$:     MOV     #2,R3           ;MAKE [DEST] = 2
3085  006614  000257                          CCC                     ;SCOPE SYNC
3086
3087  006616  062703  000002          2$:     ADD     #2,R3           ;TEST THE ADD
3088
3089  006622  022703  000004                  CMP     #4,R3           ;RESULT = 4 ?
3090  006626  001402                          BEQ     TST73           ;;BR IF YES
3091
3092  006630  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3093  006632  000766                          BR      1$              ;LOCK ON HARD ERROR
3094
3095                                  ;;**********************************************************************
3096                                  ;*TEST 73        BASIC "ADD  #N,(R) " TEST
3097                                  ;;**********************************************************************
3098  006634                          TST73:
3099  006634  012700  000073                  MOV     #73,R0          ;;LOAD R0 WITH TEST NUMBER
3100  006640  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3101  006644  012704  000004                  MOV     #4,R4           ;RESULT S / B = 4
3102  006650  012712  000002          1$:     MOV     #2,(R2)         ;MAKE [DEST] = 2
```

```
3103  006654  000257                          CCC                     ;SCOPE SYNC
3104
3105  006656  062712  000002          2$:     ADD     #2,(R2) ;TEST THE ADD
3106
3107  006662  020412                          CMP     R4,(R2)         ;RESULT = 4 ?
3108  006664  001402                          BEQ     TST74           ;;BR IF YES
3109
3110  006666  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3111  006670  000767                          BR      1$              ;LOCK ON HARD ERROR
3112
3113                                  ;;**********************************************************************
3114                                  ;*TEST 74        BASIC "ADD  #N,X(R)" TEST
3115                                  ;;**********************************************************************
3116  006672                          TST74:
3117  006672  012700  000074                  MOV     #74,R0          ;;LOAD R0 WITH TEST NUMBER
3118  006676  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
3119  006702  012702  063314                  MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0 + 2
3120  006706  012705  063312          1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
3121  006712  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
3122  006714  000257                          CCC                     ;SCOPE SYNC
3123
3124  006716  062765  000002  000002  2$:     ADD     #2,2(R5)        ;TEST THE ADD
3125
3126  006724  020412                          CMP     R4,(R2)         ;RESULT = 2 ?
3127  006726  001402                          BEQ     TST75           ;;BR IF YES
3128
3129  006730  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3130  006732  000765                          BR      1$              ;LOOP ON HARD ERROR
3131
3132                                  ;;**********************************************************************
3133                                  ;*TEST 75        BASIC "CMPB #N,(SP)+" TEST
3134                                  ;;**********************************************************************
3135  006734                          TST75:
3136  006734  012700  000075                  MOV     #75,R0          ;;LOAD R0 WITH TEST NUMBER
3137  006740  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
3138  006744  010605                          MOV     SP,R5           ;SAVE SP
3139  006746  010602                          MOV     SP,R2           ;SET UP DEST ADDR
3140  006750  005742                          TST     -(R2)           ;R2 CONTAINS DEST ADDR
3141  006752  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3142  006754  010446                  MOV     R4,-(SP)        ;MAKE [DEST] = 177400
3143  006756  000257                  CCC                     ;SCOPE SYNC - "Z" = 0
3144
3145  006760  122726  000000          2$:     CMPB    #0,(SP)+        ;TEST THE CMPB
3146
3147  006764  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
3148
3149  006766  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
3150  006770  000770                          BR      1$              ;LOCK ON HARD ERROR
3151
3152  006772  020506                  4$:     CMP     R5,SP           ;DID SP GET UPDATED BY 2?
3153  006774  001402                          BEQ     6$              ;BR IF YES
3154
3155  006776  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE SP PROPERLY
3156  007000  000764                          BR      1$              ;LOCK ON HARD ERROR
3157
3158  007002  020412                  6$:     CMP     R4,(R2)         ;[DEST] ALTERED?
```

```
3159  007004  001402                        BEQ     TST76           ;;BR IF NOT
3160
3161  007006  000000              7$:       HALT                    ;CMPB MODIFIED [DEST]
3162  007010  000760                        BR      1$              ;LOCK ON HARD ERROR.
3163
3164                              ;;**************************************************************
3165                              ;*TEST 76        BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST EVEN
3166                              ;;**************************************************************
3167  007012                      TST76:
3168  007012  012700  000076                 MOV     #76,R0          ;;LOAD R0 WITH TEST NUMBER
3169  007016  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3170  007022  012702  063324                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3171  007026  012705  063330      1$:        MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3172  007032  010203                         MOV     R2,R3           ;R3 GETS DEST ADDR
3173  007034  000257                         CCC                     ;SCOPE SYNC
3174
3175  007036  122523              2$:       CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3176
3177  007040  001402                        BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3178
3179  007042  000000              3$:       HALT                    ;CMPB FAILED TO SET "Z"
3180  007044  000770                        BR      1$              ;LOCK ON HARD ERROR
3181
3182  007046  022703  063325      4$:       CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
3183  007052  001402                        BEQ     6$              ;BR IF YES
3184
3185  007054  000000              5$:       HALT                    ;CMPB FAILED TO UPDATE DEST REG
3186  007056  000763                        BR      1$              ;LOCK ON HARD ERROR
3187
3188  007060  022705  063331      6$:       CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3189  007064  001402                        BEQ     8$              ;BR IF YES
3190
3191  007066  000000              7$:       HALT                    ;CMPB FAILED TO UPDATE SRC REG
3192  007070  000756                        BR      1$              ;LOCK ON HARD ERROR
3193
3194  007072  020412              8$:       CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3195  007074  001403                        BEQ     TST77           ;;BR IF NOT
3196
3197  007076  000000              9$:       HALT                    ;CMPB DELIVERED A RESULT
3198  007100  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
3199  007102  000751                        BR      1$              ;LOCK ON HARD ERROR
3200
3201                              ;;**************************************************************
3202                              ;*TEST 77        BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST ODD
3203                              ;;**************************************************************
3204  007104                      TST77:
3205  007104  012700  000077                 MOV     #77,R0          ;;LOAD R0 WITH TEST NUMBER
3206  007110  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3207  007114  012702  063324                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3208  007120  012705  063327      1$:        MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
3209  007124  012703  063325                 MOV     #DWTA+3,R3      ;R3 GETS DEST ADDR+1
3210  007130  000257                         CCC                     ;SCOPE SYNC
3211
3212  007132  122523              2$:       CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3213
3214  007134  001402                        BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
```

```
3215
3216  007136  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
3217  007140  000767                          BR      1$              ;LOCK ON HARD ERROR
3218
3219  007142  022703  063326          4$:     CMP     #DWTA+4,R3      ;DID DEST REG GET UPDATED?
3220  007146  001402                          BEQ     6$              ;BR IF YES
3221
3222  007150  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3223  007152  000762                          BR      1$              ;LOCK ON HARD ERROR
3224
3225  007154  022705  063330          6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3226  007160  001402                          BEQ     8$              ;BR IF YES
3227
3228  007162  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3229  007164  000755                          BR      1$              ;LOCK ON HARD ERROR
3230
3231  007166  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3232  007170  001403                          BEQ     TST100          ;;BR IF NOT
3233
3234  007172  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
3235  007174  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
3236  007176  000750                          BR      1$              ;LOCK ON HARD ERROR
3237
3238                                  ;;**********************************************************
3239                                  ;*TEST 100      BASIC "CMPB (RA)+,(RB)+" - SRC / EVEN,DEST / ODD
3240                                  ;;**********************************************************
3241  007200                          TST100:
3242  007200  012700  000100                  MOV     #100,R0         ;;LOAD R0 WITH TEST NUMBER
3243  007204  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
3244  007210  012702  063326                  MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
3245  007214  012705  063330          1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3246  007220  012703  063327                  MOV     #DWTA+5,R3      ;R3 GETS DEST ADDR
3247  007224  000257                          CCC                     ;SCOPE SYNC
3248
3249  007226  122523                  2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3250
3251  007230  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3252
3253  007232  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
3254  007234  000767                          BR      1$              ;LOCK ON HARD ERROR
3255
3256  007236  022703  063330          4$:     CMP     #DWTA+6,R3      ;DID DEST REG GET UPDATED?
3257  007242  001402                          BEQ     6$              ;BR IF YES
3258
3259  007244  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3260  007246  000762                          BR      1$              ;LOCK ON HARD ERROR
3261
3262  007250  022705  063331          6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3263  007254  001402                          BEQ     8$              ;BR IF YES
3264
3265  007256  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3266  007260  000755                          BR      1$              ;LOCK ON HARD ERROR
3267
3268  007262  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3269  007264  001403                          BEQ     TST101          ;;BR IF NOT
3270
```

```
3271  007266  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
3272  007270  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
3273  007272  000750                          BR      1$              ;LOCK ON HARD ERROR
3274
3275                                  ;;**********************************************************************
3276                                  ;*TEST 101       BASIC "CMPB (RA)+,(RB)+" - SRC / ODD,DEST / EVEN
3277                                  ;;**********************************************************************
3278  007274                          TST101:
3279  007274  012700  000101                 MOV     #101,R0         ;;LOAD R0 WITH TEST NUMBER
3280  007300  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3281  007304  012702  063324                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3282  007310  012705  063327          1$:    MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
3283  007314  010203                          MOV     R2,R3           ;R3 GETS DEST ADDR
3284  007316  000257                          CCC                     ;SCOPE SYNC
3285
3286  007320  122523                  2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3287
3288  007322  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3289
3290  007324  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
3291  007326  000770                          BR      1$              ;LOCK ON HARD ERROR
3292
3293  007330  022703  063325          4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
3294  007334  001402                          BEQ     6$              ;BR IF YES
3295
3296  007336  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3297  007340  000763                          BR      1$              ;LOCK ON HARD ERROR
3298
3299  007342  022705  063330          6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3300  007346  001402                          BEQ     8$              ;BR IF YES
3301
3302  007350  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3303  007352  000756                          BR      1$              ;LOCK ON HARD ERROR
3304
3305  007354  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3306  007356  001403                          BEQ     TST102          ;;BR IF NOT
3307
3308  007360  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
3309  007362  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
3310  007364  000751                          BR      1$              ;LOCK ON HARD ERROR
3311
3312                                  ;;**********************************************************************
3313                                  ;*TEST 102       BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN
3314                                  ;;**********************************************************************
3315  007366                          TST102:
3316  007366  012700  000102                 MOV     #102,R0         ;;LOAD R0 WITH TEST NUMBER
3317  007372  012702  063316                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3318  007376  012703  063312                 MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3319  007402  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 177400
3320  007406  012705  064630          1$:    MOV     #DBTA,R5        ;SRC ADDR = DBTA
3321  007412  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
3322  007416  000257                          CCC                     ;SCOPE SYNC
3323
3324  007420  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3325
3326  007424  020412                          CMP     R4,(R2)         ;RESULT OK?
```

```
3327  007426  001402                        BEQ     4$              ;BR IF YES
3328
3329  007430  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3330  007432  000765                        BR      1$              ;LOCK ON HARD ERROR
3331
3332  007434  022705  064631        4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
3333  007440  001402                        BEQ     TST103          ;;BR IF YES
3334
3335  007442  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3336  007444  000760                        BR      1$              ;LOCK ON HARD ERROR
3337                                ;;*****************************************************************
3338                                ;*TEST 103      BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST ODD
3339                                ;;*****************************************************************
3340  007446                        TST103:
3341  007446  012700  000103                MOV     #103,R0         ;;LOAD R0 WITH TEST NUMBER
3342  007452  012702  063316                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3343  007456  012703  063312                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3344  007462  012704  000777                MOV     #777,R4         ;RESULT 5 / B = 777
3345  007466  012705  064635        1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3346  007472  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
3347  007476  000257                        CCC                     ;SCOPE SYNC
3348
3349  007500  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3350
3351  007504  020412                        CMP     R4,(R2)         ;RESULT OK?
3352  007506  001402                        BEQ     4$              ;BR IF YES
3353
3354  007510  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3355  007512  000765                        BR      1$              ;LOCK ON HARD ERROR
3356
3357  007514  022705  064636        4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3358  007520  001402                        BEQ     TST104          ;;BR IF YES
3359
3360  007522  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3361  007524  000760                        BR      1$              ;LOCK ON HARD ERROR
3362                                ;;*****************************************************************
3363                                ;*TEST 104      BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST ODD
3364                                ;;*****************************************************************
3365  007526                        TST104:
3366  007526  012700  000104                MOV     #104,R0         ;;LOAD R0 WITH TEST NUMBER
3367  007532  012702  063316                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3368  007536  012703  063312                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3369  007542  012704  000377                MOV     #377,R4         ;RESULT 5 / B = 377
3370  007546  012705  064630        1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
3371  007552  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
3372  007556  000257                        CCC                     ;SCOPE SYNC
3373
3374  007560  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3375
3376  007564  020412                        CMP     R4,(R2)         ;RESULT OK?
3377  007566  001402                        BEQ     4$              ;BR IF YES
3378
3379  007570  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3380  007572  000765                        BR      1$              ;LOCK ON HARD ERROR
3381
3382  007574  022705  064631        4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
```

```
3393  007600  001402                        BEQ     TST105          ;;BR IF YES
3394
3395  007602  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3396  007604  000760                        BR      1$              ;LOCK ON HARD ERROR
3397                                ;;**************************************************************
3398                                ;*TEST 105      BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST EVEN
3399                                ;;**************************************************************
3400  007606                        TST105:
3401  007606  012700  000105               MOV     #105,R0         ;;LOAD R0 WITH TEST NUMBER
3402  007612  012702  063316               MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3403  007616  012703  063312               MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3404  007622  012704  177401               MOV     #177401,R4      ;RESULT S / B = 177401
3405  007626  012705  064635       1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3406  007632  012712  177777               MOV     #-1,(R2)        ;[DEST] = 177777
3407  007636  000257                       CCC                     ;SCOPE SYNC
3408
3409  007640  112563  000004       2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3410
3411  007644  020412                       CMP     R4,(R2)         ;RESULT OK?
3412  007646  001402                       BEQ     4$              ;BR IF YES
3413
3414  007650  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3415  007652  000765                       BR      1$              ;LOCK ON HARD ERROR
3416
3417  007654  022705  064636       4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3418  007660  001402                       BEQ     TST106          ;;BR IF YES
3419
3420  007662  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3421  007664  000760                       BR      1$              ;LOCK ON HARD ERROR
3422
3423                                ;;**************************************************************
3424                                ;*TEST 106      BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST EVEN
3425                                ;;**************************************************************
3426  007666                        TST106:
3427  007666  012700  000106               MOV     #106,R0         ;;LOAD R0 WITH TEST NUMBER
3428  007672  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3429  007676  012704  177401               MOV     #177401,R4      ;RESULT S / B = 177401
3430  007702  012705  064032               MOV     #DWTB,R5        ;SRC ADDR = DWTB
3431  007706  010203                1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3432  007710  012713  177777               MOV     #-1,(R3)        ;[DEST] = 177400
3433  007714  000257                       CCC                     ;SCOPE SYNC
3434
3435  007716  116523  000002       2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3436
3437  007722  020412                       CMP     R4,(R2)         ;RESULT OK?
3438  007724  001402                       BEQ     4$              ;BR IF YES
3439
3440  007726  000000                3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3441  007730  000766                       BR      1$              ;LOCK ON HARD ERROR
3442
3443  007732  022703  063313       4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3444  007736  001402                       BEQ     TST107          ;;BR IF YES
3445
3446  007740  000000                5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3447  007742  000761                       BR      1$              ;LOCK ON HARD ERROR
3448
```

```
3439                                    ;;*****************************************************************
3440                                    ;*TEST 107      BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST EVEN
3441                                    ;;*****************************************************************
3442  007744                            TST107:
3443  007744  012700  000107                   MOV     #107,R0         ;;LOAD R0 WITH TEST NUMBER
3444  007750  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3445  007754  012704  177401                   MOV     #177401,R4      ;RESULT S / B = 177401
3446  007760  012705  064634                   MOV     #DBTB,R5        ;SRC ADDR = DBTB
3447  007764  010203            1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3448  007766  012713  177777                   MOV     #-1,(R3)        ;[DEST] = 177777
3449  007772  000257                           CCC                     ;SCOPE SYNC
3450
3451  007774  116523  000001          2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3452
3453  010000  020412                           CMP     R4,(R2)         ;RESULT OK?
3454  010002  001402                           BEQ     4$              ;BR IF YES
3455
3456  010004  000000            3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3457  010006  000766                           BR      1$              ;LOCK ON HARD ERROR
3458
3459  010010  022703  063313      4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3460  010014  001402                           BEQ     TST110          ;;BR IF YES
3461
3462  010016  000000            5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3463  010020  000761                           BR      1$              ;LOCK ON HARD ERROR
3464
3465                                    ;;*****************************************************************
3466                                    ;*TEST 110      BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST ODD
3467                                    ;;*****************************************************************
3468  010022                            TST110:
3469  010022  012700  000110                   MOV     #110,R0         ;;LOAD R0 WITH TEST NUMBER
3470  010026  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3471  010032  012704  000777                   MOV     #777,R4         ;RESULT S / B = 777
3472  010036  012705  064032                   MOV     #DWTB,R5        ;SRC ADDR = DWTB
3473  010042  012703  063313      1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR
3474  010046  012712  177777                   MOV     #-1,(R2)        ;[DEST] = 177777
3475  010052  000257                           CCC                     ;SCOPE SYNC
3476
3477  010054  116523  000002          2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3478
3479  010060  020412                           CMP     R4,(R2)         ;RESULT OK?
3480  010062  001402                           BEQ     4$              ;BR IF YES
3481
3482  010064  000000            3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3483  010066  000765                           BR      1$              ;LOCK ON HARD ERROR
3484
3485  010070  022703  063314      4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3486  010074  001402                           BEQ     TST111          ;;BR IF YES
3487
3488  010076  000000            5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3489  010100  000760                           BR      1$              ;LOCK ON HARD ERROR
3490
3491                                    ;;*****************************************************************
3492                                    ;*TEST 111      BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST ODD
3493                                    ;;*****************************************************************
3494  010102                            TST111:
```

```
3495  010102  012700  000111          MOV     #111,R0         ;;LOAD R0 WITH TEST NUMBER
3496  010106  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3497  010112  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3498  010116  012705  064634          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3499  010122  012703  063313    1$:   MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR = MBUF0+1
3500  010126  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3501  010132  000257                  CCC                     ;SCOPE SYNC
3502
3503  010134  116523  000001    2$:   MOVB    1(R5),(R3)+     ;TEST THE MOVB
3504
3505  010140  020412                  CMP     R4,(R2)         ;RESULT OK?
3506  010142  001402                  BEQ     4$              ;BR IF YES
3507
3508  010144  000000    3$:           HALT                    ;MOVB DELIVERED WRONG RESULT
3509  010146  000765                  BR      1$              ;LOCK ON HARD ERROR
3510
3511  010150  022703  063314    4$:   CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3512  010154  001402                  BEQ     TST112          ;;BR IF YES
3513
3514  010156  000000    5$:           HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3515  010160  000760                  BR      1$              ;LOCK ON HARD ERROR
3516
3517                    ;;**************************************************************
3518                    ;*TEST 112      BASIC "MOVB -(RA),RB" TEST - SRC EVEN ADDR
3519                    ;;**************************************************************
3520  010162            TST112:
3521  010162  012700  000112          MOV     #112,R0         ;;LOAD R0 WITH TEST NUMBER
3522  010166  012705  063331    1$:   MOV     #DWTA+7,R5      ;SRC ADDR = DWTA+7
3523  010172  005003                  CLR     R3              ;[DEST] = 000000
3524  010174  000257                  CCC                     ;SCOPE SYNC
3525
3526  010176  114503    2$:           MOVB    -(R5),R3        ;TEST THE MOVB
3527
3528  010200  022703  177777          CMP     #-1,R3          ;RESULT OK?
3529  010204  001402                  BEQ     4$              ;BR IF YES
3530
3531  010206  000000    3$:           HALT                    ;MOVB FAILED - WRONG RESULT
3532  010210  000766                  BR      1$              ;LOCK ON HARD ERROR
3533
3534  010212  022705  063330    4$:   CMP     #DWTA+6,R5      ;SRC REG GET DECREMENTED?
3535  010216  001402                  BEQ     TST113          ;;BR IF YES
3536
3537  010220  000000    5$:           HALT                    ;MOVB FAILED TO UPDATE SRC REG
3538  010222  000761                  BR      1$              ;LOCK ON HARD ERROR
3539                    ;;**************************************************************
3540                    ;*TEST 113      BASIC "MOVB -(RA),RB" TEST - SRC 000 ADDR
3541                    ;;**************************************************************
3542  010224            TST113:
3543  010224  012700  000113          MOV     #113,R0         ;;LOAD R0 WITH TEST NUMBER
3544  010230  012705  063330    1$:   MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3545  010234  005003                  CLR     R3              ;[DEST] = 000000
3546  010236  000257                  CCC                     ;SCOPE SYNC
3547
3548  010240  114503    2$:           MOVB    -(R5),R3        ;TEST THE MOVB
3549
3550  010242  022703  177777          CMP     #-1,R3          ;RESULT OK?
```

```
3551  010246  001402                    BEQ     4$              ;BR IF YES
3552
3553  010250  000000              3$:   HALT                    ;MOVB FAILED - WRONG RESULT
3554  010252  000766                    BR      1$              ;LOCK ON HARD ERROR
3555
3556  010254  022705  063327      4$:   CMP     #DWTA+5,R5      ;SRC REG GET DECREMENTED?
3557  010260  001402                    BEQ     TST114          ;;BR IF YES
3558
3559  010262  000000              5$:   HALT                    ;MOVB FAILED TO UPDATE SRC REG
3560  010264  000761                    BR      1$              ;LOCK ON HARD ERROR
3561
3562                              ;;*****************************************************************
3563                              ;*TEST 114      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR EVEN
3564                              ;;*****************************************************************
3565  010266                      TST114:
3566  010266  012700  000114            MOV     #114,R0         ;;LOAD R0 WITH TEST NUMBER
3567  010272  010605                    MOV     SP,R5           ;SAVE SP
3568  010274  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
3569  010300  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3570  010302  012703  064630            MOV     #DBTA,R3        ;SRC ADDR = DBTA
3571  010306  012746  177777            MOV     #-1,-(SP)       ;(DEST) = 177777
3572  010312  010602                    MOV     SP,R2           ;R2 GETS DEST ADDR
3573  010314  005726                    TST     (SP)+           ;RESET SP
3574  010316  000257                    CCC                     ;SCOPE SYNC
3575
3576  010320  112346              2$:   MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3577
3578  010322  022703  064631            CMP     #DBTA+1,R3      ;DID MOVB INCREMENT SRC REG?
3579  010326  001402                    BEQ     4$              ;BR IF YES
3580
3581  010330  000000              3$:   HALT                    ;MOVB FAILED TO UPDATE SRC REG
3582  010332  000762                    BR      1$              ;LOCK ON HARD ERROR
3583
3584  010334  020412              4$:   CMP     R4,(R2)         ;RESULT OK?
3585  010336  001402                    BEQ     6$              ;BR IF YES
3586
3587  010340  000000                    HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3588  010342  000756                    BR      1$              ;LOCK ON HARD ERROR
3589
3590  010344  020206              6$:   CMP     R2,SP           ;DID SP GET PUSHED BY 2 ?
3591  010346  001402                    BEQ     8$              ;BR IF YES
3592
3593  010350  000000                    HALT                    ;MOVB FAILED TO PUSH SP PROPERLY
3594  010352  000752                    BR      1$              ;LOCK ON HARD ERROR
3595
3596  010354  010506              8$:   MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3597
3598                              ;;*****************************************************************
3599                              ;*TEST 115      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR ODD
3600                              ;;*****************************************************************
3601  010356                      TST115:
3602  010356  012700  000115            MOV     #115,R0         ;;LOAD R0 WITH TEST NUMBER
3603  010362  010605                    MOV     SP,R5           ;SAVE SP
3604  010364  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
3605  010370  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3606  010372  012703  064035            MOV     #DWTB+3,R3      ;SRC ADDR = DWTB+3
```

```
3607  010376  012746  177777          MOV     #-1,-(SP)       ;[DEST] = 177777
3608  010402  010602                  MOV     SP,R2           ;R2 GETS DEST ADDR
3609  010404  005726                  TST     (SP)+           ;RESET SP
3610  010406  000257                  CCC                     ;SCOPE SYNC
3611
3612  010410  112346          2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3613
3614  010412  022703  064036          CMP     #DWTB+4,R3      ;DID MOVB INCREMENT SRC REG?
3615  010416  001402                  BEQ     4$              ;BR IF YES
3616
3617  010420  000000          3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3618  010422  000762                  BR      1$              ;LOCK ON HARD ERROR
3619
3620  010424  020412          4$:     CMP     R4,(R2)         ;RESULT OK?
3621  010426  001402                  BEQ     6$              ;BR IF YES
3622
3623  010430  000000                  HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3624  010432  000756                  BR      1$              ;LOCK ON HARD ERROR
3625
3626  010434  020206          6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2
3627  010436  001402                  BEQ     8$              ;BR IF YES
3628
3629  010440  000000                  HALT                    ;MOVB FAILED TO PUSH SP
3630  010442  000752                  BR      1$              ;LOCK ON HARD ERROR
3631
3632  010444  010506          8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3633
3634                          ;;#####################################################################
3635                          ;*TEST 116      BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST EVEN
3636                          ;;#####################################################################
3637  010446                  TST116:
3638  010446  012700  000116          MOV     #116,R0         ;;LOAD R0 WITH TEST NUMBER
3639  010452  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3640  010456  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
3641  010462  012705  064032          MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3642  010466  005012          1$:     CLR     (R2)            ;[DEST] = 000000
3643  010470  000257                  CCC                     ;SCOPE SYNC
3644
3645  010472  116537  000006  063312  2$:     MOVB    6(R5),@#MBUF0   ;TEST THE MOVB
3646
3647  010500  020412                  CMP     R4,(R2)         ;RESULT OK?
3648  010502  001402                  BEQ     TST117          ;;BR IF YES
3649
3650  010504  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3651  010506  000767                  BR      1$              ;LOCK ON HARD ERRROR
3652                          ;;#####################################################################
3653                          ;*TEST 117      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST EVEN
3654                          ;;#####################################################################
3655  010510                  TST117:
3656  010510  012700  000117          MOV     #117,R0         ;;LOAD R0 WITH TEST NUMBER
3657  010514  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3658  010520  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
3659  010524  012705  064634          MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3660  010530  005012          1$:     CLR     (R2)            ;[DEST] = 000000
3661  010532  000257                  CCC                     ;SCOPE SYNC
3662
```

```
3663  010534  116537  000001  063312  2$:     MOVB    1(R5),@#MBUF0   ;TEST THE MOVB
3664
3665  010542  020412                          CMP     R4,(R2)         ;RESULT OK?
3666  010544  001402                          BEQ     TST120          ;;BR IF YES
3667
3668  010546  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3669  010550  000767                          BR      1$              ;LOCK ON HARD ERRROR
3670                                  ;;**************************************************************
3671                                  ;*TEST 120      BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST ODD
3672                                  ;;**************************************************************
3673  010552                          TST120:
3674  010552  012700  000120                  MOV     #120,R0         ;;LOAD R0 WITH TEST NUMBER
3675  010556  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3676  010562  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3677  010566  012705  064032                  MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3678  010572  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3679  010574  000257                          CCC                     ;SCOPE SYNC
3680
3681  010576  116537  000006  063313  2$:     MOVB    6(R5),@#MBUF0+1 ;TEST THE MOVB
3682
3683  010604  020412                          CMP     R4,(R2)         ;RESULT OK?
3684  010606  001402                          BEQ     TST121          ;;BR IF YES
3685
3686  010610  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3687  010612  000767                          BR      1$              ;LOCK ON HARD ERRROR
3688                                  ;;**************************************************************
3689                                  ;*TEST 121      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST ODD
3690                                  ;;**************************************************************
3691  010614                          TST121:
3692  010614  012700  000121                  MOV     #121,R0         ;;LOAD R0 WITH TEST NUMBER
3693                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT2
3694  010620  032737  000004  063234          BIT     #BIT2,@#BPTLOC  ;BREAKPOINT HALT SET ??
3695  010626  001401                          BEQ     .+4             ;BR IF NOT
3696  010630  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
3697  010632  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3698  010636  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3699  010642  012705  064634                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3700  010646  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3701  010650  000257                          CCC                     ;SCOPE SYNC
3702
3703  010652  116537  000001  063313  2$:     MOVB    1(R5),@#MBUF0+1 ;TEST THE MOVB
3704
3705  010660  020412                          CMP     R4,(R2)         ;RESULT OK?
3706  010662  001402                          BEQ     TST122          ;;BR IF YES
3707
3708  010664  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3709  010666  000767                          BR      1$              ;LOCK ON HARD ERRROR
3710
3711                                  ;;**************************************************************
3712                                  ;*TEST 122      BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=0
3713                                  ;;**************************************************************
3714  010670                          TST122:
3715  010670  012700  000122                  MOV     #122,R0         ;;LOAD R0 WITH TEST NUMBER
3716  010674  000257                  1$:     CCC                     ;CLEAR ALL FLAGS
3717
3718  010676  001404                  2$:     BEQ     3$              ;NO BR SHOULD OCCUR-FLAG=0
```

```
3719  010700  100403                  BMI     3$              ;NO BR SHOULD OCCUR-FLAG=0
3720  010702  102402                  BVS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3721  010704  103401                  BCS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3722  010706  000402                  BR      TST123          ;;GO TO NEXT TEST
3723
3724  010710  000000          3$:     HALT                    ;ONE OF ABOVE BR'S FAILED
3725  010712  000770                  BR      1$              ;ERROR LOOP RETURN
3726
3727                          ;;**********************************************************************
3728                          ;*TEST 123      BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLAG=1
3729                          ;;**********************************************************************
3730  010714                  TST123:
3731  010714  012700  000123          MOV     #123,R0         ;;LOAD R0 WITH TEST NUMBER
3732  010720  000277          1$:     SCC                     ;MAKE N:C = 1111
3733
3734  010722  001402          21$:    BEQ     22$             ;TEST THE BEQ-IT SHOULD BR
3735
3736  010724  000000          3$:     HALT                    ;BEQ FAILED
3737  010726  000774                  BR      1$              ;ERROR LOOP RETURN
3738
3739  010730  100402          22$:    BMI     23$             ;TEST THE BMI-IT SHOULD BR
3740
3741  010732  000000          5$:     HALT                    ;BMI FAILED
3742  010734  000771                  BR      1$              ;ERROR LOOP RETURN
3743
3744  010736  102402          23$:    BVS     24$             ;TEST THE BVS-IT SHOULD BR
3745
3746  010740  000000          7$:     HALT                    ;BVS FAILED
3747  010742  000766                  BR      1$              ;ERROR LOOP RETURN
3748
3749  010744                  24$:
3750  010744  103402                  BCS     TST124          ;;TEST THE BCS-IT SHOULD BR
3751
3752  010746  000000          9$:     HALT                    ;BCS FAILED
3753  010750  000763                  BR      1$              ;ERROR LOOP RETURN
3754
3755                          ;;**********************************************************************
3756                          ;*TEST 124      BASIC BVC TEST WITH V=1
3757                          ;;**********************************************************************
3758  010752                  TST124:
3759  010752  012700  000124          MOV     #124,R0         ;;LOAD R0 WITH TEST NUMBER
3760
3761  010756  000262          1$:     SEV                     ;MAKE V=1
3762
3763  010760  102001          2$:     BVC     3$              ;TEST THE BVC-IT SHOULDN'T BR
3764  010762  000402                  BR      TST125          ;;GO TO NEXT TEST
3765
3766  010764  000000          3$:     HALT                    ;BVC FAILED
3767  010766  000773                  BR      1$              ;ERROR LOOP RETURN
3768
3769                          ;;**********************************************************************
3770                          ;*TEST 125      BASIC BVC TEST WITH V=0
3771                          ;;**********************************************************************
3772  010770                  TST125:
3773  010770  012700  000125          MOV     #125,R0         ;;LOAD R0 WITH TEST NUMBER
3774
```

```
3775  010774  000242                  1$:     CLV                     ;MAKE V=0
3776
3777  010776                          2$:
3778  010776  102002                          BVC     TST126          ;;TEST THE BVC-IT SHOULD BR
3779
3780  011000  000000                  3$:     HALT                    ;BVC FAILED
3781  011002  000774                          BR      1$              ;ERROR LOOP RETURN
3782
3783                                  ;;**********************************************************************
3784                                  ;*TEST 126       BASIC BGE TEST WITH N,V = 00
3785                                  ;;**********************************************************************
3786  011004                          TST126:
3787  011004  012700  000126                 MOV     #126,R0         ;;LOAD R0 WITH TEST NUMBER
3788
3789  011010  000257                  1$:     CCC                     ;MAKE N:C = 0000
3790
3791  011012                          2$:
3792  011012  002002                          BGE     TST127          ;;TEST THE BGE-IT SHOULD BR
3793
3794  011014  000000                  3$:     HALT                    ;BGE FAILED
3795  011016  000774                          BR      1$              ;ERROR LOOP RETURN
3796
3797                                  ;;**********************************************************************
3798                                  ;*TEST 127       BASIC BGE TEST WITH N,V = 01
3799                                  ;;**********************************************************************
3800  011020                          TST127:
3801  011020  012700  000127                 MOV     #127,R0         ;;LOAD R0 WITH TEST NUMBER
3802
3803  011024  000257                  1$:     CCC                     ;CLEAR FLAGS
3804  011026  000262                          SEV                     ;MAKE N,V = 01
3805
3806  011030  002001                  2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3807  011032  000402                          BR      TST130          ;;GO TO NEXT TEST
3808
3809  011034  000000                  3$:     HALT                    ;BGE FAILED
3810  011036  000772                          BR      1$              ;ERROR LOOP RETURN
3811
3812                                  ;;**********************************************************************
3813                                  ;*TEST 130       BASIC BGE TEST WITH N,V = 10
3814                                  ;;**********************************************************************
3815  011040                          TST130:
3816  011040  012700  000130                 MOV     #130,R0         ;;LOAD R0 WITH TEST NUMBER
3817
3818  011044  000257                  1$:     CCC                     ;CLEAR FLAGS
3819  011046  000270                          SEN                     ;MAKE N,V = 10
3820
3821  011050  002001                  2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3822  011052  000402                          BR      TST131          ;;GO TO NEXT TEST
3823
3824  011054  000000                  3$:     HALT                    ;BGE FAILED
3825  011056  000772                          BR      1$              ;ERROR LOOP RETURN
3826
3827                                  ;;**********************************************************************
3828                                  ;*TEST 131       BASIC BGE TEST WITH N,V = 11
3829                                  ;;**********************************************************************
3830  011060                          TST131:
```

```
3831  011060  012700  000131          MOV     #131,R0         ;;LOAD R0 WITH TEST NUMBER
3832
3833  011064  000257          1$:     CCC                     ;CLEAR FLAGS
3834  011066  000272                  272                     ;MAKE N,V = 11
3835
3836  011070                  2$:
3837  011070  002002                  BGE     TST132          ;;TEST THE BGE-IT SHOULD BR
3838
3839  011072  000000          3$:     HALT                    ;BGE FAILED
3840  011074  000773                  BR      1$              ;ERROR LOOP RETURN
3841
3842                          ;;**************************************************************
3843                          ;*TEST 132      BASIC BLT TEST WITH N,V = 00
3844                          ;;**************************************************************
3845  011076                  TST132:
3846  011076  012700  000132          MOV     #132,R0         ;;LOAD R0 WITH TEST NUMBER
3847
3848  011102  000257          1$:     CCC                     ;CLEAR FLAGS
3849
3850  011104  002401          2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3851  011106  000402                  BR      TST133          ;;GO TO NEXT TEST
3852
3853  011110  000000          3$:     HALT                    ;BLT FAILED
3854  011112  000773                  BR      1$              ;ERROR LOOP RETURN
3855
3856                          ;;**************************************************************
3857                          ;*TEST 133      BASIC BLT TEST WITH N,V = 01
3858                          ;;**************************************************************
3859  011114                  TST133:
3860  011114  012700  000133          MOV     #133,R0         ;;LOAD R0 WITH TEST NUMBER
3861
3862  011120  000257          1$:     CCC                     ;CLEAR FLAGS
3863  011122  000262                  SEV                     ;MAKE N,V = 01
3864
3865  011124                  2$:
3866  011124  002402                  BLT     TST134          ;;TEST THE BLT-IT SHOULD BR
3867
3868  011126  000000          3$:     HALT                    ;BLT FAILED
3869  011130  000773                  BR      1$              ;ERROR LOOP RETURN
3870
3871                          ;;**************************************************************
3872                          ;*TEST 134      BASIC BLT TEST WITH N,V = 10
3873                          ;;**************************************************************
3874  011132                  TST134:
3875  011132  012700  000134          MOV     #134,R0         ;;LOAD R0 WITH TEST NUMBER
3876
3877  011136  000257          1$:     CCC                     ;CLEAR FLAGS
3878  011140  000270                  SEN                     ;SET N - N,V = 10
3879
3880  011142                  2$:
3881  011142  002402                  BLT     TST135          ;;TEST THE BLT-IT SHOULD BR
3882
3883  011144  000000          3$:     HALT                    ;BLT FAILED
3884  011146  000773                  BR      1$              ;ERROR LOOP RETURN
3885
3886                          ;;**************************************************************
```

```
3887                                    ;*TEST 135      BASIC BLT TEST WITH N,V = 11
3888                                    ;;**************************************************************
3889  011150                            TST135:
3890  011150  012700  000135                   MOV     #135,R0         ;;LOAD R0 WITH TEST NUMBER
3891
3892  011154  000257                    1$:     CCC                     ;CLEAR FLAGS
3893  011156  000272                            272                     ;MAKE N,V = 11
3894
3895  011160  002401                    2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3896  011162  000402                            BR      TST136          ;;GO TO NEXT TEST
3897
3898  011164  000000                    3$:     HALT                    ;BLT FAILED
3899  011166  000772                            BR      1$              ;ERROR LOOP RETURN
3900
3901                                    ;;**************************************************************
3902                                    ;*TEST 136      BASIC BGT TEST WITH Z = 1 AND N,V = 01
3903                                    ;;**************************************************************
3904  011170                            TST136:
3905  011170  012700  000136                   MOV     #136,R0         ;;LOAD R0 WITH TEST NUMBER
3906
3907  011174  000257                    1$:     CCC                     ;CLEAR FLAGS
3908  011176  000266                            266                     ;SET Z AND V
3909
3910  011200  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3911  011202  000402                            BR      TST137          ;;GO TO NEXT TEST
3912
3913  011204  000000                    3$:     HALT                    ;BGT FAILED
3914  011206  000772                            BR      1$              ;ERROR LOOP RETURN
3915
3916                                    ;;**************************************************************
3917                                    ;*TEST 137      BASIC BGT TEST WITH Z = 0 AND N,V = 01
3918                                    ;;**************************************************************
3919  011210                            TST137:
3920  011210  012700  000137                   MOV     #137,R0         ;;LOAD R0 WITH TEST NUMBER
3921
3922  011214  000257                    1$:     CCC                     ;CLEAR FLAGS
3923  011216  000262                            SEV                     ;SET V
3924
3925  011220  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3926  011222  000402                            BR      TST140          ;;GO TO SCOPE LOOP EXIT
3927
3928  011224  000000                    3$:     HALT                    ;BGT FAILED
3929  011226  000772                            BR      1$              ;ERROR LOOP RETURN
3930
3931                                    ;;**************************************************************
3932                                    ;*TEST 140      BASIC BGT TEST WITH Z = 1 AND N,V = 00
3933                                    ;;**************************************************************
3934  011230                            TST140:
3935  011230  012700  000140                   MOV     #140,R0         ;;LOAD R0 WITH TEST NUMBER
3936
3937  011234  000257                    1$:     CCC                     ;CLEAR FLAGS
3938  011236  000264                            SEZ                     ;SET Z
3939
3940  011240  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3941  011242  000402                            BR      TST141          ;;GO TO SCOPE LOOP EXIT
3942
```

```
011244  000000          3$:     HALT                    ;BGT FAILED
011246  000772                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 141      BASIC BGT TEST WITH Z = 0 AND N,V = 00
                        ;;**********************************************************************
011250                  TST141:
011250  012700  000141          MOV     #141,R0         ;;LOAD R0 WITH TEST NUMBER

011254  000257          1$:     CCC                     ;CLEAR FLAGS

011256                  2$:
011256  003002                  BGT     TST142          ;;TEST THE BGT - IT SHOULD BR

011260  000000          3$:     HALT                    ;BGT FAILED
011262  000774                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 142      BASIC BGT TEST WITH Z = 1 AND N,V = 01
                        ;;**********************************************************************
011264                  TST142:
011264  012700  000142          MOV     #142,R0         ;;LOAD R0 WITH TEST NUMBER

011270  000257          1$:     CCC                     ;CLEAR FLAGS
011272  000266                  266                     ;MAKE N,V = 01 AND Z = 1

011274  003001          2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
011276  000402                  BR      TST143          ;;GO TO NEXT TEST

011300  000000          3$:     HALT                    ;BGT FAILED
011302  000772                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 143      BASIC BGT TEST WITH Z = 1 AND N,V = 10
                        ;;**********************************************************************
011304                  TST143:
011304  012700  000143          MOV     #143,R0         ;;LOAD R0 WITH TEST NUMBER

011310  000257          1$:     CCC                     ;CLEAR FLAGS
011312  000274                  274                     ;MAKE Z = 1 AND N,V = 10

011314  003001          2$:     BGT     3$              ;TEST THE BLT-IT SHOULDN'T BR
011316  000402                  BR      TST144          ;;GO TO NEXT TEST

011320  000000          3$:     HALT                    ;BLT FAILED
011322  000772                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 144      BASIC BGT TEST WITH Z = 1 AND N,V = 11
                        ;;**********************************************************************
011324                  TST144:
011324  012700  000144          MOV     #144,R0         ;;LOAD R0 WITH TEST NUMBER

011330  000257          1$:     CCC                     ;CLEAR FLAGS
011332  000276                  276                     ;MAKE Z = 1 AND N,V = 11
```

```
011334  003001          2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
011336  000402                  BR      TST145          ;;GO TO NEXT TEST

011340  000000          3$:     HALT                    ;BLT FAILED
011342  000772                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 145      BASIC BGT TEST WITH Z=0 AND N,V=11
                        ;;**********************************************************************
011344                  TST145:
011344  012700  000145          MOV     #145,R0         ;;LOAD R0 WITH TEST NUMBER

011350  000257          1$:     CCC                     ;CLEAR FLAGS
011352  000272                  272                     ;MAKE N:C=1010

011354                  2$:
011354  003002                  BGT     TST146          ;;TEST THE BGT - IT SHOULD BR

011356  000000          3$:     HALT                    ;BGT FAILED
011360  000773                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 146      BASIC BHI TEST WITH Z,C = 00
                        ;;**********************************************************************
011362                  TST146:
011362  012700  000146          MOV     #146,R0         ;;LOAD R0 WITH TEST NUMBER

011366  000257          1$:     CCC                     ;MAKE Z,C = 00

011370                  2$:
011370  101002                  BHI     TST147          ;;TEST THE BHI-IT SHOULD BR

011372  000000          3$:     HALT                    ;BHI FAILED
011374  000774                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 147      BASIC BHI TEST WITH Z,C = 01
                        ;;**********************************************************************
011376                  TST147:
011376  012700  000147          MOV     #147,R0         ;;LOAD R0 WITH TEST NUMBER

011402  000257          1$:     CCC                     ;CLEAR FLAGS
011404  000261                  SEC                     ;MAKE Z,C = 01

011406  101001          2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
011410  000402                  BR      TST150          ;;GO TO NEXT TEST

011412  000000          3$:     HALT                    ;BHI FAILED
011414  000772                  BR      1$              ;ERROR LOOP RETURN

                        ;;**********************************************************************
                        ;*TEST 150      BASIC BHI TEST WITH Z,C = 10
                        ;;**********************************************************************
011416                  TST150:
011416  012700  000150          MOV     #150,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
4055  011422  000257                  1$:     CCC                     ;CLEAR FLAGS
4056  011424  000264                          SEZ                     ;MAKE Z,C = 10
4057
4058  011426  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
4059  011430  000402                          BR      TST151          ;;GO TO NEXT TEST
4060
4061  011432  000000                  3$:     HALT                    ;BHI FAILED
4062  011434  000772                          BR      1$              ;ERROR LOOP RETURN
4063
4064                                  ;;**************************************************************
4065                                  ;*TEST 151       BASIC BHI TEST WITH Z,C = 11
4066                                  ;;**************************************************************
4067  011436                          TST151:
4068  011436  012700  000151                MOV     #151,R0         ;;LOAD R0 WITH TEST NUMBER
4069
4070  011442  000257                  1$:     CCC                     ;CLEAR FLAGS
4071  011444  000265                          265                     ;MAKE Z,C = 11
4072
4073  011446  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULDN'T BR
4074  011450  000402                          BR      TST152          ;;GO TO NEXT TEST
4075
4076  011452  000000                  3$:     HALT                    ;BHI FAILED
4077  011454  000772                          BR      1$              ;ERROR LOOP RETURN
4078
4079                                  ;;**************************************************************
4080                                  ;*TEST 152       BASIC NEG MODE 0 TEST : [DEST] GT 0
4081                                  ;;**************************************************************
4082  011456                          TST152:
4083  011456  012700  000152                MOV     #152,R0         ;;LOAD R0 WITH TEST NUMBER
4084  011462  012704  177776                MOV     #-2,R4          ;RESULT S / B = 177776
4085  011466  012703  000002          1$:   MOV     #2,R3           ;INITIAL [DEST] = 2
4086  011472  000257                        CCC                     ;CLEAR FLAGS
4087  011474  000266                        266                     ;MAKE N:C = 0110
4088
4089  011476  005403                  2$:   NEG     R3              ;TEST THE NEG
4090
4091  011500  100003                        BPL     3$
4092  011502  001402                        BEQ     3$              ;DID N:C = 1001?
4093  011504  102401                        BVS     3$
4094  011506  103402                        BCS     4$
4095
4096  011510  000000                  3$:   HALT                    ;NEGATE FAILED TO ALTER CODES PROPERLY
4097  011512  000765                        BR      1$              ;ERROR LOOP RETURN
4098
4099  011514  020304                  4$:   CMP     R3,R4           ;CORRECT RESULT?
4100  011516  001402                        BEQ     TST153          ;;BR IF YES
4101
4102  011520  000000                  5$:   HALT                    ;NEG DELIVERED WRONG RESULT
4103  011522  000761                        BR      1$              ;ERROR LOOP RETURN
4104
4105                                  ;;**************************************************************
4106                                  ;*TEST 153       BASIC "SUB #,@#" TEST
4107                                  ;;**************************************************************
4108  011524                          TST153:
4109  011524  012700  000153                MOV     #153,R0         ;;LOAD R0 WITH TEST NUMBER
4110  011530  012704  000002                MOV     #2,R4           ;RESULT S / B = 2
```

```
4111  011534  012702  063312            MOV     #MBUF0,R2       ;R2 POINTS TO DEST
4112  011540  012712  000004     1$:    MOV     #4,(R2)         ;INITIAL [DEST] = 4
4113  011544  000257                     CCC                     ;CLEAR FLAGS
4114
4115  011546  162737  000002  063312 2$:    SUB     #2,@#MBUF0      ;TEST THE SUB
4116
4117  011554  020412                     CMP     R4,(R2)         ;RESULT=2?
4118  011556  001403                     BEQ     TST154          ;;BR IF YES
4119  011560  011203                     MOV     (R2),R3         ;GET WAS DATA
4120  011562  000000           3$:       HALT                    ;SUB DELIVERED WRONG RESULT
4121  011564  000765                     BR      1$              ;ERROR LOOP RETURN
4122
4123                                     ;;*****************************************************************
4124                                     ;*TEST 154      BASIC "SUB @#,RN" TEST
4125                                     ;;*****************************************************************
4126  011566                           TST154:
4127  011566  012700  000154            MOV     #154,R0         ;;LOAD R0 WITH TEST NUMBER
4128  011572  012737  000002  063312     MOV     #2,@#MBUF0      ;SRC = 2
4129  011600  012703  000004     1$:    MOV     #4,R3           ;INITIAL [DEST] = 4
4130  011604  000257                     CCC                     ;SCOPE SYNC
4131
4132  011606  163703  063312     2$:    SUB     @#MBUF0,R3      ;TEST THE SUB
4133
4134  011612  020403                     CMP     R4,R3           ;RESULT=2?
4135  011614  001402                     BEQ     TST155          ;;BR IF YES
4136
4137  011616  000000           3$:       HALT                    ;SUB DELIVERED WRONG RESULT
4138  011620  000767                     BR      1$              ;ERROR LOOP RETURN
4139
4140                                     ;;*****************************************************************
4141                                     ;*TEST 155      BASIC "RTS RN" TEST - <N:C> = 1111
4142                                     ;;*****************************************************************
4143  011622                           TST155:
4144  011622  012700  000155            MOV     #155,R0         ;;LOAD R0 WITH TEST NUMBER
4145  011626  010605                     MOV     SP,R5           ;SAVE SP
4146  011630  010506     1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4147  011632  012703  011652            MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
4148  011636  012746  177777            MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 FROM STACK
4149  011642  000277                     SCC                     ;N:C = 1111
4150
4151  011644  000203           2$:       RTS     R3              ;TEST THE RTS - GO TO 4$
4152
4153  011646  000000           3$:       HALT                    ;RTS FAILED TO LOAD THE PC
4154  011650  000767                     BR      1$              ;LOCK ON ERROR
4155
4156  011652  100003           4$:       BPL     5$              ;N:C = 1111 ?
4157  011654  001002                     BNE     5$
4158  011656  102001                     BVC     5$
4159  011660  103402                     BCS     6$
4160
4161  011662  000000           5$:       HALT                    ;RTS ALTERED CODES - CLEARED ONE
4162  011664  000761                     BR      1$              ;LOCK ON ERROR
4163
4164  011666  020327  177777     6$:    CMP     R3,#-1          ;DID R3 GET LOADED FROM STACK ?
4165  011672  001402                     BEQ     8$              ;BR IF YES
4166
```

```
4167  011674  000000                  7$:     HALT                    ;RTS FAILED TO LOAD REG
4168  011676  000754                          BR      1$              ;LOCK ON ERROR
4169
4170  011700  020506                  8$:     CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
4171  011702  001402                          BEQ     TST156          ;;BR IF YES
4172
4173  011704  000000                  9$:     HALT                    ;RTS FAILED TO POP SP
4174  011706  000750                          BR      1$              ;LOCK ON ERROR
4175
4176                                  ;;**************************************************************
4177                                  ;*TEST 156       BASIC "RTS  PC" TEST
4178                                  ;;**************************************************************
4179  011710                          TST156:
4180  011710  012700  000156                 MOV     #156,R0         ;;LOAD R0 WITH TEST NUMBER
4181  011714  010605                          MOV     SP,R5           ;SAVE THE ORIGINAL SP
4182  011716  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4183  011720  012746  011734                 MOV     #4$,-(SP)       ;PUSH NEW PC ON STACK
4184  011724  000257                          CCC                     ;SCOPE SYNC
4185
4186  011726  000207                  2$:     RTS     PC              ;TEST THE RTS - GO TO 4$
4187
4188  011730  000000                  3$:     HALT                    ;RTS FAILED TO LOAD PC
4189  011732  000771                          BR      1$              ;LOCK ON HARD ERROR
4190
4191  011734  020605                  4$:     CMP     SP,R5           ;DID SP GET POPPED ?
4192  011736  001402                          BEQ     TST157          ;;BR IF YES
4193
4194  011740  000000                  5$:     HALT                    ;RTS FAILED TO UPDATE SP
4195  011742  000765                          BR      1$              ;LOCK ON HARD ERROR
4196
4197                                  ;;**************************************************************
4198                                  ;*TEST 157       BASIC "JSR  PC,@#A" TEST
4199                                  ;;**************************************************************
4200  011744                          TST157:
4201  011744  012700  000157                 MOV     #157,R0         ;;LOAD R0 WITH TEST NUMBER
4202                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT3
4203  011750  032737  000010  063234         BIT     #BIT3,@#BPTLOC  ;BREAKPOINT HALT SET ??
4204  011756  001401                          BEQ     .+4             ;BR IF NOT
4205  011760  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
4206  011762  010605                          MOV     SP,R5           ;SAVE ORIGINAL SP
4207  011764  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4208  011766  000257                          CCC                     ;SCOPE SYNC
4209
4210  011770  004737  012000          2$:     JSR     PC,@#4$ ;TEST THE JSR - GO TO 4$
4211
4212  011774  000000                  3$:     HALT                    ;JSR FAILED TO LOAD PC
4213  011776  000772                          BR      1$              ;LOCK ON HARD ERROR
4214
4215  012000  022726  011774          4$:     CMP     #3$,(SP)+       ;DID JSR SAVE OLD PC ON STACK ?
4216  012004  001402                          BEQ     TST160          ;;BR IF YES
4217
4218  012006  000000                  5$:     HALT                    ;JSR FAILED TO SAVE OLD PC
4219  012010  000765                          BR      1$              ;LOCK ON HARD ERROR
4220
4221                                  ;;**************************************************************
4222                                  ;*TEST 160       BASIC "RTI" TEST - N:C=0000
```

```
4223                                    ;;*****************************************************************
4224  012012                            TST160:
4225  012012  012700  000160                   MOV     #160,R0         ;;LOAD R0 WITH TEST NUMBER
4226  012016  010605                            MOV     SP,R5           ;SAVE THE SP
4227  012020  010506                    1$:     MOV     R5,SP           ;RESET THE SP FOR ERROR LOOP
4228  012022  012746  000357                   MOV     #357,-(SP)      ;NEW PSW = 357
4229  012026  012746  012046                   MOV     #4$,-(SP)       ;NEW PC = 4$
4230  012032  005037  177776                   CLR     @#PSW           ;MAKE [PSW] = 000
4231  012036  000257                            CCC                     ;MAKE N:C=0000
4232
4233  012040  000002                    2$:     RTI                     ;TEST THE RTI - GO TO 4$
4234
4235  012042  000000                    3$:     HALT                    ;RTI FAILED TO LOAD PC
4236  012044  000765                            BR      1$              ;LOOP ON HARD ERROR
4237
4238  012046  013702  177776            4$:     MOV     @#PSW,R2        ;SAVE THE [PSW] IN R2
4239  012052  022702  000357                    CMP     #357,R2         ;WAS [PSW] = 357 ?
4240  012056  001404                            BEQ     6$              ;BR IF YES
4241
4242  012060  010237  177776                   MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
4243  012064  000000                    5$:     HALT                    ;RTI FAILED TO LOAD PSW
4244  012066  000754                            BR      1$              ;LOCK ON HARD ERROR
4245
4246  012070  020605                    6$:     CMP     SP,R5           ;DID SP GET UPDATED OK ?
4247  012072  001402                            BEQ     TST161          ;;BR IF YES
4248
4249  012074  000000                    7$:     HALT                    ;RTI FAILED TO UPDATE THE SP
4250  012076  000750                            BR      1$              ;LOCK ON HARD ERROR
4251
4252                                    ;;*****************************************************************
4253                                    ;;TEST 161      BASIC "RTI" TEST WITH N:C=1111
4254                                    ;;*****************************************************************
4255  012100                            TST161:
4256  012100  012700  000161                   MOV     #161,R0         ;;LOAD R0 WITH TEST NUMBER
4257  012104  010605                            MOV     SP,R5           ;SAVE THE SP IN R5
4258  012106  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4259  012110  005046                            CLR     -(SP)           ;NEW PSW = 000000
4260  012112  012746  012130                   MOV     #4$,-(SP)       ;NEW PC = 4$
4261  012116  012737  000357  177776           MOV     #357,@#PSW      ;MAKE OLD PSW = 357
4262  012124  000240                            NOP                     ;SCOPE SYNC
4263
4264  012126  000002                    2$:     RTI                     ;TEST THE RTI - GO TO 4$
4265
4266  012130  013702  177776            4$:     MOV     @#PSW,R2        ;GET THE PSW
4267  012134  022702  000000                    CMP     #0,R2           ;WAS [PSW]=000
4268  012140  001404                            BEQ     TST162          ;;BR IF YES
4269
4270  012142  010237  177776                   MOV     R2,@#PSW        ;RESTORE ERROR PSW
4271  012146  000000                    3$:     HALT                    ;RTI FAILED TO CLEAR PSW
4272  012150  000756                            BR      1$              ;LOCK ON HARD ERROR
4273
4274                                    ;;*****************************************************************
4275                                    ;;TEST 162      BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
4276                                    ;;*****************************************************************
4277  012152                            TST162:
4278  012152  012700  000162                   MOV     #162,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
4279  012156  010605                    MOV     SP,R5           ;SAVE THE SP
4280  012160  010506            1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4281  012162  012737  012220  000020    MOV     #4$,@#20        ;SET UP IOT VECTOR
4282  012170  012737  000357  000022    MOV     #357,@#22
4283  012176  012766  177777  177776    MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
4284  012204  005037  177776            CLR     @#PSW           ;MAKE [PSW] = 000
4285  012210  000257                    CCC                     ;SCOPE SYNC
4286
4287  012212  000004            2$:     IOT                     ;TEST THE IOT
4288
4289  012214  000000            3$:     HALT                    ;IOT FAILED TO LOAD PC
4290  012216  000760                    BR      1$              ;LOCK ON HARD ERROR
4291
4292  012220  013702  177776    4$:     MOV     @#PSW,R2        ;GET THE PSW
4293  012224  022702  000357            CMP     #357,R2         ;DID IOT LOAD A 357 ?
4294  012230  001404                    BEQ     6$              ;BR IF YES
4295
4296  012232  010237  177776            MOV     R2,@#PSW        ;RESTORE ERROR PSW
4297  012236  000000            5$:     HALT                    ;IOT FAILED TO LOAD PSW
4298  012240  000747                    BR      1$              ;LOCK ON HARD ERROR
4299
4300  012242  022726  012214    6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
4301  012246  001404                    BEQ     8$              ;BR IF YES
4302
4303  012250  010237  177776            MOV     R2,@#PSW        ;RESTORE ERROR PSW
4304  012254  000000            7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
4305  012256  000740                    BR      1$              ;LOCK ON HARD ERROR
4306
4307  012260  005726            8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
4308  012262  001404                    BEQ     TST163          ;;BR IF YES
4309
4310  012264  010237  177776            MOV     R2,@#PSW        ;RESTORE ERROR PSW
4311  012270  000000            9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
4312  012272  000732                    BR      1$              ;LOCK ON HARD ERROR
 313
4314                      ;;*****************************************************************
4315                      ;*TEST 163      BASIC "IOT" TEST - VERIFY LINKAGE TO SCOPE SERVICE
4316                      ;;*****************************************************************
4317  012274              TST163:
4318  012274  012700  000163            MOV     #163,R0         ;;LOAD R0 WITH TEST NUMBER
4319  012300  010605                    MOV     SP,R5           ;SAVE SP
4320  012302  010506            1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4321  012304  005037  063244            CLR     @#SCOFLG        ;TRAP SERVICE WILL COM "SCOFLG"
4322  012310  012737  061612  000020    MOV     #SCOPEA,@#20    ;SET UP IOT VECTOR
4323  012316  005037  000022            CLR     @#22
4324  012322  000257                    CCC                     ;SCOPE SYNC
4325
4326  012324  000004            2$:     SCOPE                   ;TEST THE IOT
4327
4328  012326  005137  063244            COM     @#SCOFLG        ;SCOFLG SHOULD BECOME 000000
4329  012332  001402                    BEQ     4$              ;BR IF IT DID
4330
4331  012334  000000            3$:     HALT                    ;IOT FAILED TO LINK TO SCOPE SERVICE
4332  012336  000761                    BR      1$              ;LOCK ON HARD ERROR
4333
4334  012340  010506            4$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
```

```
                                ;;*****************************************************************
                                ;*TEST 164      BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
                                ;;*****************************************************************
012342                          TST164:
012342  012700  000164                  MOV     #164,R0         ;;LOAD R0 WITH TEST NUMBER
012346  010605                          MOV     SP,R5           ;SAVE THE SP
012350  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012352  012737  012410  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
012360  012737  000357  000022          MOV     #357,@#22
012366  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
012374  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
012400  000257                          CCC                     ;SCOPE SYNC

012402  000004                  2$:     IOT                     ;TEST THE IOT

012404  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
012406  000760                          BR      1$              ;LOCK ON HARD ERROR

012410  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
012414  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
012420  001404                          BEQ     6$              ;BR IF YES

012422  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
012426  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
012430  000747                          BR      1$              ;LOCK ON HARD ERROR

012432  022726  012404          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
012436  001404                          BEQ     8$              ;BR IF YES

012440  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
012444  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
012446  000740                          BR      1$              ;LOCK ON HARD ERROR

012450  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
012452  001404                          BEQ     TST165          ;;BR IF YES

012454  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
012460  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
012462  000732                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 165      BASIC IOT TEST - VERIFY LOADING PSW WITH 000
                                ;;*****************************************************************
012464                          TST165:
012464  012700  000165                  MOV     #165,R0         ;;LOAD R0 WITH TEST NUMBER
012470  010605                          MOV     SP,R5           ;SAVE THE SP
012472  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012474  012737  012520  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
012502  005037  000022                  CLR     @#22            ;
012506  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW] = 340
012514  000277                          SCC                     ;MAKE N:C=1111

012516  000004                  2$:     IOT                     ;TEST THE IOT

012520  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE [PSW]
012524  001404                          BEQ     6$              ;BR IF [PSW] = 000
```

```
4391
4392  012526  010237  177776          MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
4393  012532  000000          3$:     HALT                    ;IOT FAILED TO CLEAR THE PSW
4394  012534  000756                  BR      1$              ;LOCK ON HARD ERROR
4395
4396  012536  010506          6$:     MOV     R5,SP           ;RESET THE SP BEFORE CONTINUING
4397
4398                          ;;*********************************************************************
4399                          ;*TEST 166       BASIC "TRAP" TEST - LINKAGE TO PRINT ROUTINE
4400                          ;;*********************************************************************
4401  012540                  ^ST166:
4402  012540  012700  000166          MOV     #166,R0         ;;LOAD R0 WITH TEST NUMBER
4403  012544  010605                  MOV     SP,R5           ;SAVE THE SP
4404  012546  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4405  012550  005037  063236          CLR     @#PRIFLG        ;INITIALIZE TEST FLAG
4406  012554  005037  000036          CLR     @#36            ;SET UP THE "TRAP" VECTOR
4407  012560  012737  062202  000034  MOV     #PRINA,@#34
4408  012566  000257                  CCC                     ;SCOPE SYNC
4409
4410  012570  104401          2$:     TYPE                    ;TEST THE TRAP
4411
4412  012572  012737  063166  000034  MOV     #^TRAP,@#34     ;SETUP TRAP VECTOR
4413  012600  012737  000340  000036  MOV     #340,@#36
4414  012606  005137  063236          COM     @#PRIFLG        ;SHOULD MAKE (PRIFLG) = 000000
4415  012612  001402                  BEQ     ^TST167         ;;BR IF IT DID
4416
4417  012614  000000          3$:     HALT                    ;TRAP FAILED TO LINK TO PRINT SERV.
4418  012616  000753                  BR      1$              ;LOCK ON HARD ERROR
4419
4420                          ;;*********************************************************************
4421                          ;*TEST 167       BASIC "EMT" TEST - LINKAGE TO ERROR SERVICE
4422                          ;;*********************************************************************
4423  012620                  ^ST167:
4424  012620  012700  000167          MOV     #167,R0         ;;LOAD R0 WITH TEST NUMBER
4425  012624  010605                  MOV     SP,R5           ;SAVE THE SP
4426  012626  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4427  012630  012737  062040  000030  MOV     #^ERRA,@#30     ;SET UP THE EMT VECTOR
4428  012636  005037  000032          CLR     @#32
4429  012642  005037  063240          CLR     @#ERRFLG        ;EMT SERVICE WILL COM (ERRFLG)
4430  012646  000257                  CCC                     ;SCOPE SYNC
4431
4432  012650  104000          2$:     ERROR                   ;TEST THE EMT
4433
4434  012652  005137  063240          COM     @#ERRFLG        ;DID EMT SERV. COM ERRFLG?
4435  012656  001402                  BEQ     ^TST170         ;;BR IF YES
4436
4437  012660  000000          3$:     HALT                    ;EMT DID NOT LINK PROPERLY
4438  012662  000761                  BR      1$              ;LOCK ON HARD ERROR
4439                          ;;*********************************************************************
4440                          ;*TEST 170       BASIC TEST OF RSVD INSTR. TRAP LINKAGE
4441                          ;;*********************************************************************
4442  012664                  ^ST170:
4443  012664  012700  000170          MOV     #170,R0         ;;LOAD R0 WITH TEST NUMBER
4444  012670  010605                  MOV     SP,R5           ;SAVE THE SP
4445  012672  012737  061114  000010  MOV     #^RSVTST,@#10   ;SET UP RSVD INSTR. TRAP VECTOR
4446  012700  012737  000340  000012  MOV     #340,@#12
```

```
4447  012706  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4448  012710  005037  063246                  CLR     @#RSVFLG        ;INITIALIZE TEST FLAG THAT WILL GET
4449                                                                    ;COMPLEMENTED BY TRAP SERVICE
4450  012714  000257                          CCC                     ;SCOPE SYNC
4451
4452  012716  000007                  2$:     000007                  ;FORCE RSVD INSTR. TRAP
4453
4454  012720  005137  063246                  COM     @#RSVFLG                ;TEST FLAG SHOULD GO TO 000000
4455  012724  001402                          BEQ     4$              ;BR IF TRAP SPRUNG
4456
4457  012726  000000                  3$:     HALT                    ;RSVD INSTR. TRAP FAILED
4458  012730  000766                          BR      1$              ;LOCK ON HARD ERROR
4459
4460  012732  012737  061122  000010  4$:     MOV     #RSERR,@#10     ;SET UP RSVD INSTR TRAP VECTOR TO POINT
4461  012740  012737  000340  000012          MOV     #340,@#12       ;TO ERROR SERVICE ROUTINE
4462
4463                                  ;;**********************************************************************
4464                                  ;*TEST 171       BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
4465                                  ;;**********************************************************************
4466  012746                          ;ST171:
4467  012746  012700  000171                  MOV     #171,R0         ;;LOAD R0 WITH TEST NUMBER
4468  012752  010605                          MOV     SP,R5           ;SAVE THE SP
4469  012754  012737  061212  000004          MOV     #BETST,@#4      ;SET UP THE BUS ERROR VECTOR
4470  012762  012737  000340  000006          MOV     #340,@#6
4471  012770  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4472  012772  005037  063250                  CLR     @#BERFLG        ;INITIALIZE TEST FLAG THAT WILL GET
4473                                                                    ;COMPLEMENTED BY TRAP SERVICE
4474  012776  000257                          CCC                     ;SCOPE SYNC
4475
4476  013000  005737  177700          2$:     TST     @#177700        ;FORCE BUS TIMEOUT USING R0 ADDR.
4477
4478  013004  005137  063250                  COM     @#BERFLG        ;TEST FLAG SHOULD GO TO 000000
4479  013010  001402                          BEQ     TST172          ;;BR IF TRAP SPRUNG
4480
4481  013012  000000                  3$:     HALT                    ;BUS ERROR FAILED TO SPRING TRAP
4482  013014  000765                          BR      1$              ;LOCK ON HARD ERROR
4483
4484                                  ;;**********************************************************************
4485                                  ;*TEST 172       BASIC TEST FOR ACCESSING DL11 REGISTERS
4486                                  ;;**********************************************************************
4487  013016                          ;ST172:
4488  013016  012700  000172                  MOV     #172,R0         ;;LOAD R0 WITH TEST NUMBER
4489  013022  005067  050264                  CLR     MBUF0           ;INIT STALL COUNTER
4490  013026  005367  050260          11$:    DEC     MBUF0           ;COUNT THE TIMER
4491  013032  001375                          BNE     11$             ;BR IF NO TIMEOUT
4492  013034  012737  013074  000004          MOV     #3$,@#4 ;SET UP BUS TIMEOUT VECTOR
4493  013042  012737  000340  000006          MOV     #340,@#6
4494  013050  010605                          MOV     SP,R5           ;SAVE TH SP
4495  013052  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4496  013054  012702  177560                  MOV     #RCSR,R2        ;[R2] = STARTING DL11 ADDR.
4497  013060  000257                          CCC                     ;SCOPE SYNC
4498
4499  013062  005722                  2$:     TST     (R2)+           ;REFERENCE DL11 - RCSR
4500  013064  005722                          TST     (R2)+           ;REFERENCE DL11 - RDBR
4501  013066  005722                          TST     (R2)+           ;REFERENCE DL11 - XCSR
4502  013070  005712                          TST     (R2)            ;REFERENCE DL11 - XDBR
```

```
                                                BR      4$              ;GO TO NEXT TEST

                                3$:             TST     -(R2)           ;BAD ADDRESS IN R2
                                                HALT                    ;ONE OF DL11 ADDR'S CAUSED TIME OUT
                                                BR      1$              ;LOCK ON HARD ERROR

                                4$:             MOV     #BERR,@#4       ;SET UP BUS ERROR VECTOR TO POINT
                                                MOV     #340,@#6        ;TO ERROR SERVICE ROUTINE
                                ;;**************************************************************
                                ;*TEST 173      BASIC TEST OF DL11 - XCSR - READY(1)
                                ;;**************************************************************
                                TST173:
                                                MOV     #173,R0         ;;LOAD R0 WITH TEST NUMBER
                                                MOV     #XCSR,R2        ;DEST ADDR = XCSR
                                                MOV     #200,R4         ;RESULT S / B = 200
                                1$:             CLR     (R2)            ;CLEAR [DEST]
                                                CLR     R1              ;SET UP TIMEOUT COUNTER
                                                CCC                     ;SCOPE SYNC

                                2$:             CMP     R4,(R2)         ;TEST READY BIT - IT SHOULD BE SET

                                                BEQ     TST174          ;;BR IF IT WAS
                                                DEC     R1              ;TICK-TOCK GOES THE TIMER
                                                BNE     2$              ;BR IF NOT A TIMEOUT

                                                MOV     (R2),R3         ;GET THE WAS DATA
                                3$:             HALT                    ;READY BIT IN XCSR FAILED ON A (0)
                                                BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 174      BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
                                ;;**************************************************************
                                TST174:
                                                MOV     #174,R0         ;;LOAD R0 WITH TEST NUMBER
                                                MOV     #XCSR,R2        ;DEST ADDR = XCSR
                                                MOV     #200,R4         ;RESULT S / B = 200
                                1$:             CLR     (R2)            ;CLEAR MAINT. BIT
                                                CCC                     ;SCOPE SYNC

                                2$:             CMP     R4,(R2)         ;TEST MAINT(0)

                                                BEQ     TST175          ;;BR IF MAINT BIT CLEAR

                                                MOV     (R2),R3         ;GET THE WAS DATA
                                3$:             HALT                    ;CAN'T CLEAR MAINT BIT
                                                BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 175      BASIC TEST OF DL11 XCSR - MAINT BIT = 1
                                ;;**************************************************************
                                TST175:
                                                MOV     #175,R0         ;;LOAD R0 WITH TEST NUMBER
                                                MOV     #XCSR,R2        ;DEST ADDR = XCSR
                                                MOV     #204,R4         ;RESULT S / B = 204
                                1$:             MOV     #4,(R2)         ;SET THE MAINT. BIT
```

```
4559  013230  000257                          CCC                     ;SCOPE SYNC
4560
4561  013232  020412                  2$:     CMP     R4,(R2)         ;TEST MAINT.(1)
4562
4563  013234  001403                          BEQ     TST176          ;;BR IF IT WAS
4564
4565  013236  011203                          MOV     (R2),R3         ;GET THE WAS DATA
4566  013240  000000                  3$:     HALT                    ;CAN'T SET MAINT BIT IN XCSR
4567  013242  000770                          BR      1$              ;LOCK 0 HARD ERROR
4568
4569                                  ;;*****************************************************************
4570                                  ;*TEST 176       BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
4571                                  ;THIS ROUTINE USES THE MAINTENANCE MODE FEATURE OF THE DL11 TO
4572                                  ;TURN AROUND A STRING OF 8 CHARACTERS TO THE DL11. THIS STRING CONSISTS
4573                                  ;OF ALTERNATING NULL / DELETE CHARS WHICH ARE NON PRINTING. THE 8 CHARS
4574                                  ;ARE OUTPUT THEN READ BACK INTO A CORE BUFFER AND THEN THE INPUT AND
4575                                  ;OUTPUT CORE BUFFERS ARE CHECKED FOR EQUIVALENCE. IF AN ERROR IS DET-
4576                                  ;ECTED DURING THE COMPARISON THE ROUTINE HALTS WITH THE WAS AND S / B
4577                                  ;DATA IN R3 AND R4 RESPECTFULLY. A TIMER IS EMPLOYED TO PREVENT THE
4578                                  ;TEST FROM HANGING IF RECEIVER DONE DOES NOT RESPOND.
4579                                  ;*****************************************************************
4580  013244                          TST176:
4581  013244  012700  000176                  MOV     #176,R0         ;;LOAD R0 WITH TEST NUMBER
4582  013250  012702  177560          6$:     MOV     #RCSR,R2        ;R2 POINTS TO DL11 - START ADDR
4583  013254  105762  000002                  TSTB    2(R2)           ;REFERENCE DL11 INPUT DATA BUFFER TWICE
4584  013260  105762  000002                  TSTB    2(R2)           ;TO FLUSH RCVR "DONE" BIT
4585  013264  012703  063266                  MOV     #IBUF,R3        ;R3 POINTS TO CORE INPUT BUFFER
4586  013270  012704  063256                  MOV     #OBUF,R4        ;R4 POINTS TO CORE OUTPUT BUFFER
4587  013274  012705  000010                  MOV     #10,R5          ;R5 WILL COUNT 8 CHARS OUTPUT
4588  013300  012762  000004  000004          MOV     #4,4(R2)        ;TURN ON MAINT MODE
4589
4590  013306  005001                  1$:     CLR     R1              ;R1 USED AS TIMEOUT COUNTER
4591  013310  112462  000006                  MOVB    (R4)+,6(R2)     ;LOAD OUTPUT BUFFER IN DL11
4592  013314  105712                  2$:     TSTB    (R2)            ;RECEIVER DONE SET ?
4593  013316  100404                          BMI     3$              ;BR IF YES
4594  013320  005301                          DEC     R1              ;COUNT THE TIMER
4595  013322  001374                          BNE     2$              ;BR IF NO TIMEOUT
4596
4597  013324  000000                          HALT                    ;DL11 FAILED TO RESPOND IN TIME
4598  013326  000750                          BR      6$              ;LOCK ON HARD ERROR
4599
4600  013330  116223  000002          3$:     MOVB    2(R2),(R3)+     ;READ THE DL11 INPUT BUFFER INTO CORE
4601  013334  005305                          DEC     R5              ;COUNT ONE CHAR
4602  013336  001363                          BNE     1$              ;BR IF NOT DONE 8 CHARS
4603
4604  013340  005062  000004                  CLR     4(R2)           ;TURN OFF MAINT. MODE
4605  013344  012705  000010                  MOV     #10,R5          ;RESET CHAR COUNTER
4606  013350  012703  063266                  MOV     #IBUF,R3        ;RESET INBUF POINTER
4607  013354  012704  063256                  MOV     #OBUF,R4        ;RESET OUTBUF POINTER
4608
4609  013360  122324                  4$:     CMPB    (R3)+,(R4)+     ;INPUT = OUTPUT ??
4610  013362  001003                          BNE     5$              ;BR IF NOT
4611  013364  005305                          DEC     R5              ;COUNT ONE CHECKED
4612  013366  001374                          BNE     4$              ;BR UNTIL 8 DONE
4613  013370  000410                          BR      CITST           ;GO TO NEXT TEST
4614
```

```
4615  013372  114303            5$:   MOVB    -(R3),R3        ;WAS DATA IN R3 (BITS 7:0)
4616  013374  114404                  MOVB    -(R4),R4        ;S / B DATA IN R4 (BITS 7:0)
4617  013376  042703  177400          BIC     #177400,R3      ;STRIP OFF BITS (15:08)
4618  013402  042704  177400          BIC     #177400,R4      ;  "           "
4619  013406  000000                  HALT                    ;RECEIVED DATA NOT EQUAL TO OUTPUT DATA
4620  013410  000717                  BR      5$              ;LOCK ON HARD ERROR
```

```
4621                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4622                                    ;////////////COMPREHENSIVE INSTRUCTION TESTS/////////////////////////
4623                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4624
4625  013412  012737  061260  000020  CITST:  MOV     #$SCOPE,@#20    ;SET UP IOT VECTOR
4626  013420  005037  000022                  CLR     @#22
4627  013424  012737  061620  000030          MOV     #$ERROR,@#30    ;SET UP EMT VECTOR
4628  013432  012737  000340  000032          MOV     #340,@#32
4629  013440  012737  063166  000034          MOV     #$TRAP,@#34     ;SET UP TRAP VECTOR
4630  013446  012737  000340  000036          MOV     #340,@#36
4631  013454  012737  060664  000024          MOV     #$PWRDN,@#24    ;SET UP POWER FAIL VECTOR
4632  013462  012737  000340  000026          MOV     #340,@#26
4633  013470  105737  001141                  TSTB    @#SENVM         ;DO NOT SIZE BIT SET?
4634  013474  100003                          BPL     3$              ;BR IF NOT - USE HARDWARE SWITCH REG
4635  013476  012737  001142  001040          MOV     #$SWREG,@#SWR   ;USE APT SWITCH REG.
4636  013504  032777  010000  165326  3$:     BIT     #SW12,@SWR      ;INHIBIT PRINTING INTRO. I.D. MESSAGE?
4637  013512  001007                          BNE     1$              ;BR IF YES
4638  013514  005737  063254                  TST     @#ONCE          ;FIRST TIME INTO "CIT" TESTS ?
4639  013520  001004                          BNE     1$              ;BR IF NOT - PRINT ID ONLY ONCE
4640  013522  005137  063254                  COM     @#ONCE          ;SET FLAG TO INHIBIT PRINTING AGAIN
4641  013526  104401                          TYPE                    ;IDENTIFY THIS PROGRAM
4642  013530  065141                          IDENT1                  ;ADDR OF THE ID MESSAGE
4643  013532  005037  177776          1$:     CLR     @#PSW           ;SET CPU PRIORITY TO LEVEL 000
4644  013536  012737  003316  001006          MOV     #TST0,@#SLPADR  ;INITIALIZE SCOPE LOOP RETURN
4645  013544  012737  000040  001110          MOV     #40,@#STIMES    ;ITERATE ON BIT SECTION 32 TIMES
4646  013552  010037  001124                  MOV     R0,@#STESTN     ;PREVENT MISSED TEST ERROR ON
4647                                                                  ;FIRST SCOPE CALL
4648
4649                                    ;;*****************************************************************
4650                                    ;*TEST 177      BCC TEST WITH C=1
4651                                    ;;*****************************************************************
4652  013556                            TST177:
4653  013556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4654  013560  012700  000177                  MOV     #177,R0         ;;LOAD R0 WITH TEST NUMBER
4655  013564  013701  013572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4656  013570  000261                          SEC                     ;MAKE C=1
4657
4658  013572  103001                  2$:     BCC     3$              ;TEST THE BCC, IT SHOULDN'T BR
4659  013574  000401                          BR      TST200          ;;GO TO SCOPE EXIT
4660
4661  013576  104005                  3$:     ERROR   5               ;BCC FAILED
4662
4663                                    ;;*****************************************************************
4664                                    ;*TEST 200      BCC TEST WITH C=0
4665                                    ;;*****************************************************************
4666  013600                            TST200:
4667  013600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4668  013602  012700  000200                  MOV     #200,R0         ;;LOAD R0 WITH TEST NUMBER
4669  013606  013701  013614                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4670  013612  000241                          CLC                     ;MAKE C=0
4671
4672  013614                          2$:
4673  013614  103001                          BCC     TST201          ;;TEST THE BCC-IT SHOULD BR
4674
4675  013616  104005                  3$:     ERROR   5               ;BCC FAILED
4676
```

```
                                ;;*****************************************************************
                                ;*TEST 201      VERIFY NO BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013620                          TST201:
013620  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013622  012700  000201                  MOV     #201,R0         ;;LOAD R0 WITH TEST NUMBER
013626  013701  013644                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013632  012704  000017                  MOV     #17,R4          ;S/B PSW
013636  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013642  000277                          SCC                     ;MAKE N:C = 1111

013644  103004                  2$:     BCC     3$              ;TEST THE BCC-IT SHOULDN'T BR

013646  013703  177776                  MOV     @#PSW,R3        ;GET WAS FLAGS
013652  020304                          CMP     R3,R4           ;N:C = 1111?
013654  001401                          BEQ     TST202          ;;BR IF YES

013656  104001                  3$:     ERROR   1               ;NO BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 202      VERIFY BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013660                          TST202:
013660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013662  012700  000202                  MOV     #202,R0         ;;LOAD R0 WITH TEST NUMBER
013666  013701  013704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013672  012704  000017                  MOV     #17,R4          ;S/B PSW
013676  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013702  000277                          SCC                     ;MAKE N:C = 1111

013704  000401                  2$:     BR      4$              ;TEST THE BR

013706  104005                  3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK

013710  013703  177776          4$:     MOV     @#PSW,R3        ;GET THE FLAGS
013714  020304                          CMP     R3,R4           ;N:C = 1111?
013716  001401                          BEQ     TST203          ;;BR IF YES

013720  104001                  5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 203      VERIFY NO BRANCH MICROROUTINE DOES NOT SET FLAGS
                                ;;*****************************************************************
013722                          TST203:
013722  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013724  012700  000203                  MOV     #203,R0         ;;LOAD R0 WITH TEST NUMBER
013730  013701  013744                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013734  005004                          CLR     R4              ;PSW S/B = 0
013736  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013742  000257                          CCC                     ;MAKE N:C = 0000

013744  103404                  2$:     BCS     3$              ;TEST THE BCS-IT SHOULDN'T BR
```

```
4733  013746  013703  177776          MOV     @#PSW,R3        ;GET FLAGS
4734  013752  005703                  TST     R3              ;N:C = 0000
4735  013754  001401                  BEQ     TST204          ;;BR IF YES
4736
4737  013756  104001          3$:     ERROR   1               ;NO BRANCH MICROROUTINE-ALTERED CODES
4738
4739                          ;;**********************************************************************
4740                          ;*TEST 204      VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
4741                          ;;**********************************************************************
4742  013760                  TST204:
4743  013760  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4744  013762  012700  000204          MOV     #204,R0         ;;LOAD R0 WITH TEST NUMBER
4745  013766  013701  014002          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4746  013772  005004                  CLR     R4              ;PSW S/B = 0
4747  013774  012702  177776          MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL
4748
4749  014000  000257                  CCC                     ;MAKE N:C = 0000
4750
4751  014002  000401          2$:     BR      4$              ;TEST THE BR
4752
4753  014004  104005          3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK
4754
4755  014006  013703  177776  4$:     MOV     @#PSW,R3        ;GET FLAGS
4756  014012  005703                  TST     R3              ;N:C = 0000
4757  014014  001401                  BEQ     TST205          ;;BR IF YES
4758
4759  014016  104001          5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES.
4760
4761                          ;;**********************************************************************
4762                          ;*TEST 205      BLE TEST WITH Z = 0, AND N,V = 00
4763                          ;;**********************************************************************
4764  014020                  TST205:
4765  014020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4766  014022  012700  000205          MOV     #205,R0         ;;LOAD R0 WITH TEST NUMBER
4767  014026  013701  014034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4768  014032  000257                  CCC                     ;CLEAR FLAGS
4769
4770  014034  003401          2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
4771  014036  000401                  BR      TST206          ;;GO TO SCOPE EXIT
4772
4773  014040  104005          3$:     ERROR   5               ;BLE FAILED
4774
4775                          ;;**********************************************************************
4776                          ;*TEST 206      BLE TEST WITH Z = 1 AND N,V = 00
4777                          ;;**********************************************************************
4778  014042                  TST206:
4779  014042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4780  014044  012700  000206          MOV     #206,R0         ;;LOAD R0 WITH TEST NUMBER
4781  014050  013701  014060          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4782  014054  000257                  CCC                     ;CLEAR FLAGS
4783  014056  000264                  SEZ                     ;SET Z = 1
4784
4785  014060                  2$:
4786  014060  003401                  BLE     TST207          ;;TEST THE BLE-IT SHOULD BR
4787
4788  014062  104005          3$:     ERROR   5               ;BLE FAILED
```

```
                                ;;*****************************************************************
                                ;*TEST 207      BLE TEST WITH Z = 0 AND N,V = 01
                                ;;*****************************************************************
014064                          TST207:
014064  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014066  012700  000207                  MOV     #207,R0         ;;LOAD R0 WITH TEST NUMBER
014072  013701  014102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014076  000257                          CCC                     ;CLEAR FLAGS
014100  000262                          SEV                     ;MAKE Z = 0 AND N,V = 01
                                
014102                          2$:
014102  003401                          BLE     TST210          ;;TEST THE BLE-IT SHOULD BR
                                
014104  104005                  3$:     ERROR   5               ;BLE FAILED
                                
                                ;;*****************************************************************
                                ;*TEST 210      BLE TEST WITH Z = 0 AND N,V = 10
                                ;;*****************************************************************
014106                          TST210:
014106  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014110  012700  000210                  MOV     #210,R0         ;;LOAD R0 WITH TEST NUMBER
014114  013701  014124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014120  000257                          CCC                     ;CLEAR FLAGS
014122  000270                          SEN                     ;MAKE Z = 0 AND N,V = 10
                                
014124                          2$:
014124  003401                          BLE     TST211          ;;TEST THE BLE-IT SHOULD BR
                                
014126  104005                  3$:     ERROR   5               ;BLE FAILED
                                
                                ;;*****************************************************************
                                ;*TEST 211      BLE TEST WITH Z = 0 AND N,V = 11
                                ;;*****************************************************************
014130                          TST211:
014130  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014132  012700  000211                  MOV     #211,R0         ;;LOAD R0 WITH TEST NUMBER
014136  013701  014146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014142  000257                          CCC                     ;CLEAR FLAGS
014144  000272                          272                     ;MAKE Z = 0 AND N,V = 11
                                
014146  003401                  2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
014150  000401                          BR      TST212          ;;GO TO SCOPE EXIT
                                
014152  104005                  3$:     ERROR   5               ;BLE FAILED
                                
                                ;;*****************************************************************
                                ;*TEST 212      BLOS TEST WITH Z,C = 00
                                ;;*****************************************************************
014154                          TST212:
014154  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014156  012700  000212                  MOV     #212,R0         ;;LOAD R0 WITH TEST NUMBER
014162  013701  014170                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014166  000257                          CCC                     ;MAKE Z,C = 00
                                
014170  101401                  2$:     BLOS    3$              ;TEST THE BLOS-IT SHOULDN'T BR
```

```
4845  014172  000401                    BR      TST213          ;;GO TO SCOPE EXIT
4846
4847  014174  104005                    3$:     ERROR   5               ;BLOS FAILED
4848
4849                                    ;;**********************************************************************
4850                                    ;*TEST 213      BLOS TEST WITH Z,C = 01
4851                                    ;;**********************************************************************
4852  014176                            TST213:
4853  014176  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4854  014200  012700  000213                    MOV     #213,R0         ;;LOAD R0 WITH TEST NUMBER
4855  014204  013701  014214                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4856  014210  000257                            CCC                     ;CLEAR FLAGS
4857  014212  000261                            SEC                     ;MAKE Z,C = 01
4858
4859  014214                            2$:
4860  014214  101401                            BLOS    TST214          ;;TEST THE BLOS-IT SHOULD BR
4861
4862  014216  104005                    3$:     ERROR   5               ;BLOS FAILED
4863
4864                                    ;;**********************************************************************
4865                                    ;*TEST 214      BLOS TEST WITH Z,C = 10
4866                                    ;;**********************************************************************
4867  014220                            TST214:
4868  014220  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4869  014222  012700  000214                    MOV     #214,R0         ;;LOAD R0 WITH TEST NUMBER
4870  014226  013701  014236                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4871  014232  000257                            CCC                     ;CLEAR FLAGS
4872  014234  000264                            SEZ                     ;MAKE Z,C = 10
4873
4874  014236                            2$:
4875  014236  101401                            BLOS    TST215          ;;TEST THE BLOS-IT SHOULD BR
4876
4877  014240  104005                    3$:     ERROR   5               ;BLOS FAILED
4878
4879                                    ;;**********************************************************************
4880                                    ;*TEST 215      BLOS TEST WITH Z,C = 11
4881                                    ;;**********************************************************************
4882  014242                            TST215:
4883  014242  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4884  014244  012700  000215                    MOV     #215,R0         ;;LOAD R0 WITH TEST NUMBER
4885  014250  013701  014260                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4886  014254  000257                            CCC                     ;CLEAR FLAGS
4887  014256  000265                            265                     ;MAKE Z,C = 11
4888
4889  014260                            2$:
4890  014260  101401                            BLOS    TST216          ;;TEST THE BLOS-IT SHOULD BR
4891
4892  014262  104005                    3$:     ERROR   5               ;BLOS FAILED
4893
4894                                    ;;**********************************************************************
4895                                    ;*TEST 216      SXT MODE 0 TEST WITH N = 0 AND C = 1
4896                                    ;;**********************************************************************
4897  014264                            TST216:
4898  014264  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4899  014266  012700  000216                    MOV     #216,R0         ;;LOAD R0 WITH TEST NUMBER
4900  014272  013701  014310                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
4901  014276  005004                        CLR     R4              ;RESULT S / B = 0
4902  014300  012703  177777                MOV     #-1,R3          ;INITAL DEST. OP = 177777
4903  014304  000257                        CCC                     ;CLEAR CODES
4904  014306  000263                        263                     ;N:C = 0011
4905
4906  014310  006703                 2$:    SXT     R3              ;TEST THE SXT
4907
4908  014312  100403                        BMI     3$
4909  014314  001002                        BNE     3$              ;DID SXT MAKE N:C = 0101?
4910  014316  102401                        BVS     3$
4911  014320  103401                        BCS     4$
4912
4913  014322  104002                 3$:    ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4914
4915  014324  005703                 4$:    TST     R3              ;DID RESULT = 0?
4916  014326  001401                        BEQ     TST217          ;;BR IF IT DID
4917
4918  014330  104002                 5$:    ERROR   2               ;SXT DELIVERED WRONG RESULT TO R3
4919
4920                                         ;;*****************************************************************
4921                                         ;*TEST 217      SXT MODE 0 TEST WITH N = 0 AND C = 0
4922                                         ;;*****************************************************************
4923  014332                         TST217:
4924  014332  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4925  014334  012700  000217                MOV     #217,R0         ;;LOAD R0 WITH TEST NUMBER
4926  014340  013701  014366                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4927                                 .SBTTL USER CONTROLLED BREAKPOINT -- BIT4
4928  014344  032737  000020  063234        BIT     #BIT4,@#BPTLOC  ;BREAKPOINT HALT SET ??
4929  014352  001401                        BEQ     .+4             ;BR IF NOT
4930  014354  000000                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
4931
4932  014356  005004                        CLR     R4              ;RESULT S / B = 0
4933  014360  012703  177777                MOV     #-1,R3          ;INITIAL DEST OP = 177777
4934  014364  000257                        CCC                     ;CLEAR N:C
4935
4936  014366  006703                 2$:    SXT     R3              ;TEST THE SXT
4937  014370  103001                        BCC     TST220          ;;BR IF "C" STILL CLEAR
4938
4939  014372  104002                 3$:    ERROR   2               ;SXT AFFECTED "C" BIT
4940
4941                                         ;;*****************************************************************
4942                                         ;*TEST 220      SXT MODE 0 TEST WITH N = 1 AND C = 1
4943                                         ;;*****************************************************************
4944  014374                         TST220:
4945  014374  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4946  014376  012700  000220                MOV     #220,R0         ;;LOAD R0 WITH TEST NUMBER
4947  014402  013701  014416                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4948  014406  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
4949  014412  005003                        CLR     R3              ;INITIAL DEST OP = 0
4950  014414  000277                        SCC                     ;MAKE N:C = 1111
4951
4952  014416  006703                 2$:    SXT     R3              ;TEST THE SXT
4953
4954  014420  100003                        BPL     3$
4955  014422  001402                        BEQ     3$              ;N:C = 1001?
4956  014424  102401                        BVS     3$
```

```
4957  014426  103401                  BCS     4$
4958
4959  014430  104002          3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4960
4961  014432  010305          4$:     MOV     R3,R5           ;GET RESULT
4962  014434  005105                  COM     R5              ;COMPLEMENT IT-SHOULD GO TO 0
4963  014436  001401                  BEQ     TST221          ;;BR IF RESULT OF SXT = 1
4964
4965  014440  104002          5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT.
4966
4967                          ;;**************************************************************
4968                          ;*TEST 221      SXT MODE 0 TEST WITH N = 1 AND C = 0
4969                          ;;**************************************************************
4970  014442                  TST221:
4971  014442  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4972  014444  012700  000221          MOV     #221,R0         ;;LOAD R0 WITH TEST NUMBER
4973  014450  013701  014466          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4974  014454  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
4975  014460  005003                  CLR     R3              ;INITIAL DEST OP = 0
4976  014462  000257                  CCC                     ;CLEAR FLAGS
4977  014464  000276                  276                     ;MAKE N:C = 1110
4978
4979  014466  006703          2$:     SXT     R3              ;TEST THE SXT
4980  014470  103001                  BCC     TST222          ;;BR IF "C" UNAFFECTED
4981
4982  014472  104002          3$:     ERROR   2               ;SXT SET "C" BIT
4983
4984                          ;;**************************************************************
4985                          ;*TEST 222      SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
4986                          ;;**************************************************************
4987  014474                  TST222:
4988  014474  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4989  014476  012700  000222          MOV     #222,R0         ;;LOAD R0 WITH TEST NUMBER
4990  014502  013701  014524          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4991  014506  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4992  014512  005004                  CLR     R4              ;RESULT S / B = 0
4993  014514  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4994  014520  000257                  CCC                     ;CLEAR CODES
4995  014522  000263                  263                     ;MAKE N:C = 0011
4996
4997  014524  006712          2$:     SXT     (R2)            ;TEST THE SXT - DM1
4998
4999  014526  100403                  BMI     3$
5000  014530  001002                  BNE     3$              ;N:C = 0101
5001  014532  102401                  BVS     3$
5002  014534  103401                  BCS     4$
5003
5004  014536  104001          3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5005
5006  014540  005712          4$:     TST     (R2)            ;DID RESULT = 0?
5007  014542  001401                  BEQ     11$             ;BR IF YES
5008
5009  014544  104001          5$:     ERROR   1               ;SXT SHOULD HAVE ZEROED [DEST]
5010
5011  014546  012702  063312  11$:    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5012  014552  013701  014566          MOV     @#12$,R1        ;LOAD R1 WITH TEST INSTR WORD
```

```
5013  014556  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5014  014562  000257                  CCC                     ;CLEAR CODES
5015  014564  000263                  263                     ;MAKE N:C = 0011
5016
5017  014566  006722          12$:    SXT     (R2)+           ;TEST SXT - DM2
5018
5019  014570  100403                  BMI     7$              ;N:C = 0101 ?
5020  014572  001002                  BNE     7$
5021  014574  102401                  BVS     7$
5022  014576  103401                  BCS     6$
5023
5024  014600  104001          7$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5025
5026  014602  005737  063312  6$:     TST     @#MBUF0         ;DID RESULT GET ZEROED ?
5027  014606  001401                  BEQ     8$              ;BR IF YES
5028
5029  014610  104001          9$:     ERROR   1               ;SXT FAILED TO ZERO [DEST]
5030
5031  014612  020227  063314  8$:     CMP     R2,#MBUF0+2     ;WAS IT REALLY MODE 2 ?
5032  014616  001401                  BEQ     TST223          ;;BR IF YES
5033
5034  014620  104001                  ERROR   1               ;SXT FAILED TO AUTO INCREMENT
5035
5036                          ;;*****************************************************************
5037                          ;*TEST 223      SXT MODE 1 TEST WITH N = 0 AND C = 0
5038                          ;;*****************************************************************
5039  014622                  TST223:
5040  014622  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5041  014624  012700  000223          MOV     #223,R0         ;;LOAD R0 WITH TEST NUMBER
5042  014630  013701  014650          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5043  014634  005004                  CLR     R4              ;RESULT S / B = 0
5044  014636  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5045  014642  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5046  014646  000257                  CCC                     ;CLEAR "C" BIT
5047
5048  014650  006712          2$:     SXT     (R2)            ;TEST THE SXT
5049  014652  103001                  BCC     TST224          ;;BR IF "C" UNDISTURBED
5050
5051  014654  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
5052
5053                          ;;*****************************************************************
5054                          ;*TEST 224      SXT MODE 1 TEST WITH N = 1 AND C = 1
5055                          ;;*****************************************************************
5056  014656                  TST224:
5057  014656  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5058  014660  012700  000224          MOV     #224,R0         ;;LOAD R0 WITH TEST NUMBER
5059  014664  013701  014704          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5060  014670  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5061  014674  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5062  014700  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5063  014702  000277                  SCC                     ;MAKE N:C = 1111
5064
5065  014704  006712          2$:     SXT     (R2)            ;TEST THE SXT
5066
5067  014706  100003                  BPL     3$
5068  014710  001402                  BEQ     3$              ;N:C = 1001?
```

```
5069  014712  102401                  BVS     3$
5070  014714  103401                  BCS     4$
5071
5072  014716  104001          3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5073
5074  014720  021204          4$:     CMP     (R2),R4         ;RESULT = 177777?
5075  014722  001401                  BEQ     TST225          ;;BR IF YES
5076
5077  014724  104001          5$:     ERROR   1               ;SXT DELIVERED WRONG RESULT
5078
5079                          ;;**********************************************************************
5080                          ;*TEST 225       SXT MODE 1 TEST WITH N = 1 AND C = 0
5081                          ;;**********************************************************************
5082  014726                  TST225:
5083  014726  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5084  014730  012700  000225          MOV     #225,R0         ;;LOAD R0 WITH TEST NUMBER
5085  014734  013701  014756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5086  014740  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5087  014744  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5088  014750  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5089  014752  000257                  CCC                     ;CLEAR FLAGS
5090  014754  000276                  276                     ;MAKE N:C = 1110
5091
5092  014756  006712          2$:     SXT     (R2)            ;TEST THE SXT
5093  014760  103001                  BCC     TST226          ;;BR IF "C" UNAFFECTED
5094
5095  014762  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
5096
5097                          ;;**********************************************************************
5098                          ;*TEST 226       SWAB MODE 0 TEST WITH POS. RESULT
5099                          ;;**********************************************************************
5100  014764                  TST226:
5101  014764  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5102  014766  012700  000226          MOV     #226,R0         ;;LOAD R0 WITH TEST NUMBER
5103  014772  013701  015012          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5104  014776  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
5105  015002  012703  000377          MOV     #377,R3         ;INITIAL DEST OP = 377
5106  015006  000257                  CCC                     ;CLEAR FLAGS
5107  015010  000273                  273                     ;MAKE N:C = 1011
5108
5109  015012  000303          2$:     SWAB    R3              ;TEST THE SWAB
5110
5111  015014  100403                  BMI     3$
5112  015016  001002                  BNE     3$              ;N:C = 0100
5113  015020  102401                  BVS     3$
5114  015022  103001                  BCC     4$
5115
5116  015024  104002          3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
5117
5118  015026  020403          4$:     CMP     R4,R3           ;CORRECT RESULT?
5119  015030  001401                  BEQ     TST227          ;;BR IF YES
5120
5121  015032  104002          5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
5122
5123                          ;;**********************************************************************
5124                          ;*TEST 227       SWAB MODE 0 TEST WITH NEG. RESULT
```

```
5125                                    ;;*****************************************************************
5126  015034                            TST227:
5127  015034  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5128  015036  012700  000227                    MOV     #227,R0         ;;LOAD R0 WITH TEST NUMBER
5129  015042  013701  015062                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5130  015046  012704  000377                    MOV     #377,R4         ;RESULT S  /  B = 377
5131  015052  012703  177400                    MOV     #177400,R3      ;INITIAL DEST OP = 177400
5132  015056  000257                            CCC                     ;CLEAR FLAGS
5133  015060  000267                            267                     ;MAKE N:C = 0111
5134
5135  015062  000303                    2$:     SWAB    R3              ;TEST THE SWAB
5136
5137  015064  100003                            BPL     3$
5138  015066  001402                            BEQ     3$              ;DID SWAB MAKE N:C = 1000
5139  015070  102401                            BVS     3$
5140  015072  103001                            BCC     4$
5141
5142  015074  104002                    3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
5143
5144  015076  020403                    4$:     CMP     R4,R3           ;DID SWAB DELIVER CORRECT RESULT?
5145  015100  001401                            BEQ     TST230          ;;BR IF OK
5146
5147  015102  104002                    5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
5148
5149                                    ;;*****************************************************************
5150                                    ;*TEST 230      SWAB MODE 1 AND 2 TEST WITH POS. RESULT
5151                                    ;;*****************************************************************
5152  015104                            TST230:
5153  015104  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5154  015106  012700  000230                    MOV     #230,R0         ;;LOAD R0 WITH TEST NUMBER
5155  015112  013701  015136                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5156  015116  012704  177400                    MOV     #177400,R4      ;RESULT S  /  B = 177400
5157  015122  012702  063312                    MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5158  015126  012712  000377                    MOV     #377,(R2)       ;SET UP DEST OP = 377
5159  015132  000257                            CCC                     ;CLEAR FLAGS
5160  015134  000273                            273                     ;MAKE N:C = 1011
5161
5162  015136  000312                    2$:     SWAB    (R2)            ;TEST THE SWAB - DM1
5163
5164  015140  100403                            BMI     3$
5165  015142  001002                            BNE     3$              ;N:C = 0100
5166  015144  102401                            BVS     3$
5167  015146  103001                            BCC     4$
5168
5169  015150  104001                    3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5170
5171  015152  020412                    4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5172  015154  001401                            BEQ     5$              ;BR IF OK
5173
5174  015156  104001                            ERROR   1               ;SWAB DELIVERED WRONG RESULT
5175
5176  015160  013701  015200            5$:     MOV     @#20$,R1        ;LOAD R1 WITH TEST INSTR. WORD
5177  015164  012702  063312                    MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5178  015170  012712  000377                    MOV     #377,(R2)       ;[DEST] = 000377
5179  015174  000257                            CCC                     ;CLEAR FLAGS
5180  015176  000273                            273                     ;MAKE N:C = 1011
```

```
5181
5182  015200  000322                  20$:    SWAB    (R2)+           ;TEST THE SWAB - DM2
5183
5184  015202  100403                          BMI     7$              ;N:C = 0100
5185  015204  001002                          BNE     7$
5186  015206  102401                          BVS     7$
5187  015210  103001                          BCC     6$
5188
5189  015212  104001                  7$:     ERROR   1               ;SWAB FAILED TO SET CODES PROPERLY
5190
5191  015214  020437  063312          6$:     CMP     R4,@#MBUF0      ;CORRECT RESULT ?
5192  015220  001401                          BEQ     8$              ;BR IF YES
5193
5194  015222  104001                  9$:     ERROR   1               ;SWAB DELIVERED THE WRONG RESULT
5195
5196  015224  020227  063314          8$:     CMP     R2,#MBUF0+2     ;DID AUTO INCREMENT OCCUR ?
5197  015230  001401                          BEQ     TST231          ;;BR IF YES
5198
5199  015232  104001                          ERROR   1               ;SWAB FAILED TO AUTO INC REG.
5200
5201                                  ;;*****************************************************************
5202                                  ;*TEST 231       SWAB MODE 1 TEST WITH NEG. RESULT
5203                                  ;;*****************************************************************
5204  015234                          TST231:
5205  015234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5206  015236  012700  000231                  MOV     #231,R0         ;;LOAD R0 WITH TEST NUMBER
5207  015242  013701  015266                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5208  015246  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
5209  015252  012702  063312                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5210  015256  012712  177400                  MOV     #177400,(R2)    ;SET UP DEST. OP = 177400
5211  015262  000257                          CCC                     ;CLEAR FLAGS
5212  015264  000267                          267                     ;MAKE N:C = 0111
5213
5214  015266  000312                  2$:     SWAB    (R2)            ;TEST THE SWAB
5215
5216  015270  100003                          BPL     3$
5217  015272  001402                          BEQ     3$              ;N:C = 1000?
5218  015274  102401                          BVS     3$
5219  015276  103001                          BCC     4$
5220
5221  015300  104001                  3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5222
5223  015302  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5224  015304  001401                          BEQ     TST232          ;;BR IF YES
5225
5226  015306  104001                  5$:     ERROR   1               ;SWAB DELIVERED WRONG RESULT
5227
5228                                  ;;*****************************************************************
5229                                  ;*TEST 232       NEG MODE 0 TEST : [DEST] = 0
5230                                  ;;*****************************************************************
5231  015310                          TST232:
5232  015310  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5233  015312  012700  000232                  MOV     #232,R0         ;;LOAD R0 WITH TEST NUMBER
5234  015316  013701  015332                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5235  015322  005004                          CLR     R4              ;RESULT S / B = 0
5236  015324  005003                          CLR     R3              ;INITIAL [DEST] = 0
```

```
5237  015326  000257                          CCC                     ;CLEAR FLAGS
5238  015330  000273                          273                     ;MAKE N:C = 1011
5239
5240  015332  005403                  2$:     NEG     R3              ;TEST THE NEG
5241
5242  015334  100403                          BMI     3$
5243  015336  001002                          BNE     3$              ;N:C = 0100 ONLY "Z" SET?
5244  015340  102401                          BVS     3$
5245  015342  103001                          BCC     4$
5246
5247  015344  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5248
5249  015346  020304                  4$:     CMP     R3,R4           ;WAS RESULT = 0
5250  015350  001400                          BEQ     TST233          ;;BR IF YES
5251
5252  015352  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5253
5254                                  ;;*****************************************************************
5255                                  ;*TEST 233       NEG MODE 0 TEST : [DEST] LT 0
5256                                  ;;*****************************************************************
5257  015354                          TST233:
5258  015354  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5259  015356  012700  000233                  MOV     #233,R0         ;;LOAD R0 WITH TEST NUMBER
5260  015362  013701  015402                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5261  015366  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
5262  015372  012703  177776                  MOV     #-2,R3          ;INITIAL [DEST] = 177776
5263  015376  000257                          CCC                     ;CLEAR FLAGS
5264  015400  000276                          276                     ;MAKE N:C = 1110
5265
5266  015402  005403                  2$:     NEG     R3              ;TEST THE NEG
5267
5268  015404  100403                          BMI     3$
5269  015406  001402                          BEQ     3$              ;N:C = 0001?
5270  015410  102401                          BVS     3$
5271  015412  103401                          BCS     4$
5272
5273  015414  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5274
5275  015416  020304                  4$:     CMP     R3,R4           ;RESULT = 2?
5276  015420  001401                          BEQ     TST234          ;;BR IF YES
5277
5278  015422  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5279
5280                                  ;;*****************************************************************
5281                                  ;*TEST 234       NEG MODE 0 TEST : [DEST] = 100000 (8)
5282                                  ;;*****************************************************************
5283  015424                          TST234:
5284  015424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5285  015426  012700  000234                  MOV     #234,R0         ;;LOAD R0 WITH TEST NUMBER
5286  015432  013701  015450                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5287  015436  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
5288  015442  010403                          MOV     R4,R3           ;INITIAL [DEST] = 100000
5289  015444  000257                          CCC                     ;CLEAR FLAGS
5290  015446  000264                          SEZ                     ;MAKE N:C = 01000
5291
5292  015450  005403                  2$:     NEG     R3              ;TEST THE NEG
```

```
5293
5294  015452  100003                        BPL     3$
5295  015454  001402                        BEQ     3$              ;N:C = 1011?
5296  015456  102001                        BVC     3$
5297  015460  103401                        BCS     4$
5298
5299  015462  104002                3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5300
5301  015464  020304                4$:     CMP     R3,R4           ;RESULT STILL 100000?
5302  015466  001401                        BEQ     TST235          ;;BR IF YES
5303
5304  015470  104002                5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5305
5306                                ;;**************************************************************
5307                                ;*TEST 235      NEG MODE 1 TEST : [DEST] = 0
5308                                ;;**************************************************************
5309  015472                        TST235:
5310  015472  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5311  015474  012700  000235                MOV     #235,R0         ;;LOAD R0 WITH TEST NUMBER
5312  015500  013701  015520                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5313  015504  012702  063312                MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5314  015510  005004                        CLR     R4              ;RESULT S / B = 0
5315  015512  005012                        CLR     (R2)            ;INITIAL [DEST] = 0
5316  015514  000257                        CCC                     ;CLEAR FLAGS
5317  015516  000273                        273                     ;MAKE N:C = 1011
5318
5319  015520  005412                2$:     NEG     (R2)            ;TEST THE NEG
5320
5321  015522  100403                        BMI     3$
5322  015524  001002                        BNE     3$              ;N:C = 0100?
5323  015526  102401                        BVS     3$
5324  015530  103001                        BCC     4$
5325
5326  015532  104001                3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5327
5328  015534  021204                4$:     CMP     (R2),R4         ;RESULT = 0?
5329  015536  001401                        BEQ     TST236          ;;BR IF YES
5330
5331  015540  104001                5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5332
5333                                ;;**************************************************************
5334                                ;*TEST 236      NEG MODE 1  TEST : [DEST] GT 0
5335                                ;;**************************************************************
5336  015542                        TST236:
5337  015542  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5338  015544  012700  000236                MOV     #236,R0         ;;LOAD R0 WITH TEST NUMBER
5339  015550  013701  015574                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5340  015554  012702  063312                MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5341  015560  012704  177776                MOV     #-2,R4          ;RESULT S / B = 177776
5342  015564  012712  000002                MOV     #2,(R2)         ;INITIAL [DEST] = 2
5343  015570  000257                        CCC                     ;CLEAR FLAGS
5344  015572  000266                        266                     ;MAKE N:C = 0110
5345
5346  015574  005412                2$:     NEG     (2)             ;TEST THE NEG
5347
5348  015576  100003                        BPL     3$
```

```
5349  015600  001402                          BEQ     3$              ;N:C = 1001?
5350  015602  102401                          BVS     3$
5351  015604  103401                          BCS     4$
5352
5353  015606  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5354
5355  015610  021204                  4$:     CMP     (R2),R4         ;CORRECT RESULT?
5356  015612  001401                          BEQ     TST237          ;;BR IF YES
5357
5358  015614  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5359
5360                                  ;;**********************************************************************
5361                                  ;*TEST 237       NEG MODE 1 TEST : [DEST] LT 0
5362                                  ;;**********************************************************************
5363  015616                          TST237:
5364  015616  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5365  015620  012700  000237                  MOV     #237,R0         ;;LOAD R0 WITH TEST NUMBER
5366  015624  013701  015650                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5367  015630  012702  063312                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5368  015634  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
5369  015640  012712  177776                  MOV     #-2,(R2)        ;INITIAL [DEST] = 177776
5370  015644  000257                          CCC                     ;CLEAR FLAGS
5371  015646  000276                          SVF                     ;MAKE N:C = 1110
5372
5373  015650  005412                  2$:     NEG     (R2)            ;TEST THE NEG
5374
5375  015652  100403                          BMI     3$
5376  015654  001402                          BEQ     3$              ;N:C = 0001?
5377  015656  102401                          BVS     3$
5378  015660  103401                          BCS     4$
5379
5380  015662  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5381
5382  015664  021204                  4$:     CMP     (R2),R4         ;CORRECT RESULT = 2?
5383  015666  001401                          BEQ     TST240          ;;BR IF YES
5384
5385  015670  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5386
5387                                  ;;**********************************************************************
5388                                  ;*TEST 240       NEG MODE 1 TEST: [DEST] = 100000 (8)
5389                                  ;;**********************************************************************
5390  015672                          TST240:
5391  015672  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5392  015674  012700  000240                  MOV     #240,R0         ;;LOAD R0 WITH TEST NUMBER
5393  015700  013701  015722                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5394  015704  012702  063312                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5395  015710  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
5396  015714  010412                          MOV     R4,(R2)         ;INITIAL [DEST] = 100000
5397  015716  000257                          CCC                     ;CLEAR FLAGS
5398  015720  000264                          SFZ                     ;MAKE N:Z = 0100
5399
5400  015722  005412                  2$:     NEG     (R2)            ;TEST THE NEG
5401
5402  015724  100003                          BPL     3$
5403  015726  001402                          BEQ     3$              ;N:C = 1011?
5404  015730  102001                          BVC     3$
```

```
5405  015732  103401                          BCS     4$
5406
5407  015734  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5408
5409  015736  021204                  4$:     CMP     (R2),R4         ;CORRECT RESULT = 100000?
5410  015740  001401                          BEQ     TST241          ;;BR IF YES
5411
5412  015742  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5413
5414                                  ;;**************************************************************
5415                                  ;*TEST 241      ROR TEST - DMD - N:C = 1110
5416                                  ;;**************************************************************
5417  015744                          TST241:
5418  015744  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5419  015746  012700  000241                  MOV     #241,R0         ;;LOAD R0 WITH TEST NUMBER
5420  015752  013701  015772                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5421  015756  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
5422  015762  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
5423  015766  000257                          CCC                     ;CLEAR FLAGS
5424  015770  000276                          276                     ;N:C = 1111
5425
5426  015772  006003                  2$:     ROR     R3              ;TEST THE ROR
5427
5428  015774  100403                          BMI     3$              ;N:C = 0000 ?
5429  015776  001402                          BEQ     3$
5430  016000  102401                          BVS     3$
5431  016002  103001                          BCC     4$
5432
5433  016004  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5434
5435  016006  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5436  016010  001401                          BEQ     TST242          ;;BR IF YES
5437
5438  016012  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5439
5440                                  ;;**************************************************************
5441                                  ;*TEST 242      ROR TEST - DMD - N:C = 1000
5442                                  ;;**************************************************************
5443  016014                          TST242:
5444  016014  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5445  016016  012700  000242                  MOV     #242,R0         ;;LOAD R0 WITH TEST NUMBER
5446  016022  013701  016040                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5447  016026  005004                          CLR     R4              ;RESULT S / B = 000000
5448  016030  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5449  016034  000257                          CCC                     ;CLEAR FLAGS
5450  016036  000270                          SEN                     ;N:C = 1000
5451
5452  016040  006003                  2$:     ROR     R3              ;TEST THE ROR
5453
5454  016042  100403                          BMI     3$              ;N:C = 0111 ?
5455  016044  001002                          BNE     3$
5456  016046  102001                          BVC     3$
5457  016050  103401                          BCS     4$
5458
5459  016052  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5460
```

```
5461  016054  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5462  016056  001401                          BEQ     TST243          ;;BR IF YES
5463
5464  016060  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5465
5466                                  ;;**********************************************************************
5467                                  ;*TEST 243       ROR TEST - DMO - N:C = 0111
5468                                  ;;**********************************************************************
5469  016062                          TST243:
5470  016062  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5471  016064  012700  000243                  MOV     #243,R0         ;;LOAD R0 WITH TEST NUMBER
5472  016070  013701  016110                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5473  016074  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252 ,
5474  016100  012703  052525                  MOV     #52525,R3       ;[DEST] = 052525
5475  016104  000257                          CCC                     ;CLEAR FLAGS
5476  016106  000267                          267                     ;N:C = 0111
5477
5478  016110  006003                  2$:     ROR     R3              ;TEST THE ROR
5479
5480  016112  100003                          BPL     3$              ;N:C = 1001 ?
5481  016114  001402                          BEQ     3$
5482  016116  102401                          BVS     3$
5483  016120  103401                          BCS     4$
5484
5485  016122  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5486
5487  016124  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5488  016126  001401                          BEQ     TST244          ;;BR IF YES
5489
5490  016130  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5491
5492                                  ;;**********************************************************************
5493                                  ;*TEST 244       ASR TEST - DMO - N:C = 1000
5494                                  ;;**********************************************************************
5495  016132                          TST244:
5496  016132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5497  016134  012700  000244                  MOV     #244,R0         ;;LOAD R0 WITH TEST NUMBER
5498  016140  013701  016156                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5499  016144  005004                          CLR     R4              ;RESULT S / B = 000000
5500  016146  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5501  016152  000257                          CCC                     ;CLEAR FLAGS
5502  016154  000270                          SEN                     ;N:C = 1000
5503
5504  016156  006003                  2$:     ROR     R3              ;TEST THE ROR
5505
5506  016160  100403                          BMI     3$              ;N:C = 0111 ?
5507  016162  001002                          BNE     3$
5508  016164  102001                          BVC     3$
5509  016166  103401                          BCS     4$
5510
5511  016170  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5512
5513  016172  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5514  016174  001401                          BEQ     TST245          ;;BR IF YES
5515
5516  016176  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
```

```
5517
5518                                    ;;*****************************************************************
5519                                    ;*TEST 245      ASR TEST - DM0 - N:C = 0101
5520                                    ;;*****************************************************************
5521   016200                           TST245:
5522   016200  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5523   016202  012700  000245                  MOV     #245,R0         ;;LOAD R0 WITH TEST NUMBER
5524   016206  013701  016226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5525   016212  012704  152525                  MOV     #152525,R4      ;RESULT S / B = 152525
5526   016216  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
5527   016222  000257                          CCC                     ;CLEAR FLAGS
5528   016224  000265                          265                     ;N:C = 0101
5529
5530   016226  006003                   2$:    ROR     R3              ;TEST THE ROR
5531
5532   016230  100003                          BPL     3$              ;N:C = 1010 ?
5533   016232  001402                          BEQ     3$
5534   016234  102001                          BVC     3$
5535   016236  103001                          BCC     4$
5536
5537   016240  104002                   3$:    ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5538
5539   016242  020403                   4$:    CMP     R4,R3           ;CORRECT RESULT ?
5540   016244  001401                          BEQ     TST246          ;;BR IF YES
5541
5542   016246  104002                   5$:    ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5543
5544                                    ;;*****************************************************************
5545                                    ;*TEST 246      ASR TEST - DM0 - N:C = 1100
5546                                    ;;*****************************************************************
5547   016250                           TST246:
5548   016250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5549   016252  012700  000246                  MOV     #246,R0         ;;LOAD R0 WITH TEST NUMBER
5550   016256  013701  016276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5551   016262  012704  025252                  MOV     #25252,R4       ;RESULT S / B = 25252
5552   016266  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
5553   016272  000257                          CCC                     ;CLEAR FLAGS
5554   016274  000274                          274                     ;N:C = 1100
5555
5556   016276  006003                   2$:    ROR     R3              ;TEST THE ROR
5557
5558   016300  100403                          BMI     3$              ;N:C = 0011 ?
5559   016302  001402                          BEQ     3$
5560   016304  102001                          BVC     3$
5561   016306  103401                          BCS     4$
5562
5563   016310  104002                   3$:    ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5564
5565   016312  020403                   4$:    CMP     R4,R3           ;CORRECT RESULT ?
5566   016314  001401                          BEQ     TST247          ;;BR IF YES
5567
5568   016316  104002                   5$:    ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5569
5570                                    ;;*****************************************************************
5571                                    ;*TEST 247      ROR TEST - DM1 - N:C = 1110
5572                                    ;;*****************************************************************
```

```
5573  016320                        TST247:
5574  016320  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5575  016322  012700  000247                MOV     #247,R0         ;;LOAD R0 WITH TEST NUMBER
5576  016326  013701  016352                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5577  016332  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5578  016336  012704  052525                MOV     #52525,R4       ;RESULT S / B = 52525
5579  016342  012712  125252                MOV     #125252,(R2)    ;[DEST] = 125252
5580  016346  000257                        CCC                     ;CLEAR FLAGS
5581  016350  000276                        276                     ;N:C = 1110
5582
5583  016352  006012                2$:     ROR     (R2)            ;TEST THE ROR
5584
5585  016354  100403                        BMI     3$              ;N:C = 0000 ?
5586  016356  001402                        BEQ     3$
5587  016360  102401                        BVS     3$
5588  016362  103001                        BCC     4$
5589
5590  016364  104001                3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5591
5592  016366  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5593  016370  001402                        BEQ     TST250          ;;BR IF YES
5594  016372  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5595  016374  104001                5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5596
5597                                ;;**************************************************************
5598                                ;*TEST 250      ROR TEST - DM1 - N:C = 1000
5599                                ;;**************************************************************
5600  016376                        TST250:
5601  016376  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5602  016400  012700  000250                MOV     #250,R0         ;;LOAD R0 WITH TEST NUMBER
5603  016404  013701  016426                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5604  016410  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5605  016414  005004                        CLR     R4              ;RESULT S / B = 000000
5606  016416  012712  000001                MOV     #1,(R2)         ;[DEST] = 1
5607  016422  000257                        CCC                     ;CLEAR FLAGS
5608  016424  000270                        SEN                     ;N:C = 1000
5609
5610  016426  006012                2$:     ROR     (R2)            ;TEST THE ROR
5611
5612  016430  100403                        BMI     3$              ;N:C = 0111 ?
5613  016432  001002                        BNE     3$
5614  016434  102001                        BVC     3$
5615  016436  103401                        BCS     4$
5616
5617  016440  104001                3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5618
5619  016442  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5620  016444  001402                        BEQ     TST251          ;;BR IF YES
5621
5622  016446  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5623  016450  104001                5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5624
5625                                ;;**************************************************************
5626                                ;*TEST 251      ROR TEST - DM1 - N:C = 0111
5627                                ;;**************************************************************
5628  016452                        TST251:
```

```
5629  016452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5630  016454  012700  000251                  MOV     #251,R0         ;;LOAD R0 WITH TEST NUMBER
5631  016460  013701  016504                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5632  016464  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5633  016470  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
5634  016474  012712  052525                  MOV     #52525,(R2)     ;(DEST) = 52525
5635  016500  000257                          CCC                     ;CLEAR FLAGS
5636  016502  000267                          267                     ;N:C = 0111
5637
5638  016504  006012                  2$:     ROR     (R2)            ;TEST THE ROR
5639
5640  016506  100003                          BPL     3$              ;N:C = 1001 ?
5641  016510  001402                          BEQ     3$
5642  016512  102401                          BVS     3$
5643  016514  103401                          BCS     4$
5644
5645  016516  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5646
5647  016520  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5648  016522  001402                          BEQ     TST252          ;;BR IF YES
5649
5650  016524  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5651  016526  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5652
5653                                  ;;**************************************************************
5654                                  ;*TEST 252      ASR TEST - DM1 - N:C = 1000
5655                                  ;;**************************************************************
5656  016530                          TST252:
5657  016530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5658  016532  012700  000252                  MOV     #252,R0         ;;LOAD R0 WITH TEST NUMBER
5659  016536  013701  016560                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5660  016542  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5661  016546  005004                          CLR     R4              ;RESULT S  /  B = 000000
5662  016550  012712  000001                  MOV     #1,(R2)         ;(DEST) = 1
5663  016554  000257                          CCC                     ;CLEAR FLAGS
5664  016556  000270                          SEN                     ;N:C = 1000
5665
5666  016560  006012                  2$:     ROR     (R2)            ;TEST THE ROR
5667
5668  016562  100403                          BMI     3$              ;N:C = 0111 ?
5669  016564  001002                          BNE     3$
5670  016566  102001                          BVC     3$
5671  016570  103401                          BCS     4$
5672
5673  016572  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5674
5675  016574  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5676  016576  001402                          BEQ     TST253          ;;BR IF YES
5677
5678  016600  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5679  016602  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5680
5681                                  ;;**************************************************************
5682                                  ;*TEST 253      ASR TEST - DM1 - N:C = 1100
5683                                  ;;**************************************************************
5684  016604                          TST253:
```

```
5685  016604  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5686  016606  012700  000253              MOV     #253,R0         ;;LOAD R0 WITH TEST NUMBER
5687  016612  013701  016636              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5688  016616  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5689  016622  012704  025252              MOV     #25252,R4       ;RESULT S / B = 25252
5690  016626  012712  052525              MOV     #52525,(R2)     ;[DEST] = 52525
5691  016632  000257                      CCC                     ;CLEAR FLAGS
5692  016634  000274                      274                     ;N:C = 1100
5693
5694  016636  006012              2$:     ROR     (R2)            ;TEST THE ROR
5695
5696  016640  100403                      BMI     3$              ;N:C = 0011 ?
5697  016642  001402                      BEQ     3$
5698  016644  102001                      BVC     3$
5699  016646  103401                      BCS     4$
5700
5701  016650  104001              3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5702
5703  016652  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5704  016654  001402                      BEQ     TST254          ;;BR IF YES
5705
5706  016656  011203                      MOV     (R2),R3         ;GET THE WAS DATA
5707  016660  104001              5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5708
5709                              ;;**************************************************************
5710                              ;*TEST 254      ASR TEST - DM1 - N:C = 0101
5711                              ;;**************************************************************
5712  016662                      TST254:
5713  016662  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5714  016664  012700  000254              MOV     #254,R0         ;;LOAD R0 WITH TEST NUMBER
5715  016670  013701  016714              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5716  016674  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5717  016700  012704  152525              MOV     #152525,R4      ;RESULT S / B = 152525
5718  016704  012712  125252              MOV     #125252,(R2)    ;[DEST] = 125252
5719  016710  000257                      CCC                     ;CLEAR FLAGS
5720  016712  000265                      265                     ;N:C = 0101
5721
5722  016714  006012              2$:     ROR     (R2)            ;TEST THE ROR
5723
5724  016716  100003                      BPL     3$              ;N:C = 1010 ?
5725  016720  001402                      BEQ     3$
5726  016722  102001                      BVC     3$
5727  016724  103001                      BCC     4$
5728
5729  016726  104001              3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5730
5731  016730  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5732  016732  001402                      BEQ     TST255          ;;BR IF YES
5733
5734  016734  011203                      MOV     (R2),R3         ;GET THE WAS DATA
5735  016736  104001              5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5736
5737                              ;;**************************************************************
5738                              ;*TEST 255      RORB TEST - DM2 - EVEN ADDRESS
5739                              ;;**************************************************************
5740  016740                      TST255:
```

```
5741  016740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5742  016742  012700  000255                  MOV     #255,R0         ;;LOAD R0 WITH TEST NUMBER
5743  016746  013701  016772                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5744  016752  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5745  016756  012704  000177                  MOV     #177,R4         ;RESULT S / B = 177
5746  016762  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5747  016764  012712  000377                  MOV     #377,(R2)       ;[DEST] = 377
5748  016770  000257                          CCC                     ;SCOPE SYNC "C" = 0
5749
5750  016772  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5751
5752  016774  103401                          BCS     4$              ;BR IF ROR SET "C"
5753
5754  016776  104001                  3$:     ERROR   1               ;ROR FAILED TO SET "C"
5755
5756  017000  022703  063313          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5757  017004  001401                          BEQ     6$              ;BR IF YES
5758
5759  017006  104005                  5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5760
5761  017010  020412                  6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5762  017012  001402                          BEQ     TST256          ;;BR IF YES
5763
5764  017014  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5765  017016  104001                  7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5766
5767                                  ;;**********************************************************************
5768                                  ;*TEST 256       RORB TEST - DM1 - EVEN ADDRESS
5769                                  ;;**********************************************************************
5770  017020                          TST256:
5771  017020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5772  017022  012700  000256                  MOV     #256,R0         ;;LOAD R0 WITH TEST NUMBER
5773  017026  013701  017054                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5774  017032  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5775  017036  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
5776  017042  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5777  017044  012712  000376                  MOV     #376,(R2)       ;[DEST] = 376
5778  017050  000257                          CCC                     ;CLEAR FLAGS
5779  017052  000261                          SEC                     ;SCOPE SYNC - SET "C"
5780
5781  017054  106013                  2$:     RORB    (R3)            ;TEST THE RORB
5782
5783  017056  103001                          BCC     4$              ;BR IF "C" CLR - IT SHOULD BE
5784
5785  017060  104001                  3$:     ERROR   1               ;RORB FAILED TO CLR "C"
5786
5787  017062  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5788  017064  001402                          BEQ     TST257          ;;BR IF YES
5789
5790  017066  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5791  017070  104001                  5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5792
5793                                  ;;**********************************************************************
5794                                  ;*TEST 257       RORB TEST - DM2 - ODD ADDRESS
5795                                  ;;**********************************************************************
5796  017072                          TST257:
```

```
5797  017072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5798  017074  012700  000257                  MOV     #257,R0         ;;LOAD R0 WITH TEST NUMBER
5799  017100  013701  017142                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5800                                  .SBTTL USER CONTROLLED BREAKPOINT -- BITS
5801  017104  032737  000040  063234          BIT     #BIT5,@#BPTLOC  ;BREAKPOINT HALT SET ??
5802  017112  001401                          BEQ     .+4             ;BR IF NOT
5803  017114  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
5804  017116  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5805  017122  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
5806  017126  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5807  017132  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5808  017134  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
5809  017140  000257                          CCC                     ;SCOPE SYNC - "C" =0
5810
5811  017142  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5812
5813  017144  103401                          BCS     4$              ;BR IF "C" IS SET - IT SHOULD BE
5814
5815  017146  104001                  3$:     ERROR   1               ;RORB FAILED TO SET "C"
5816
5817  017150  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5818  017154  001401                          BEQ     6$              ;BR IF YES
5819
5820  017156  104005                  5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5821
5822  017160  020415                  6$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5823  017162  001402                          BEQ     TST260          ;;BR IF YES
5824
5825  017164  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5826  017166  104001                  7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5827
5828                                  ;;*****************************************************************
5829                                  ;*TEST 260      RORB TEST - DM1 - ODD ADDRESS
5830                                  ;;*****************************************************************
5831  017170                          TST260:
5832  017170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5833  017172  012700  000260                  MOV     #260,R0         ;;LOAD R0 WITH TEST NUMBER
5834  017176  013701  017226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5835  017202  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5836  017206  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
5837  017212  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5838  017216  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5839  017220  012715  177377                  MOV     #177377,(R5)    ;[DEST] = 177377
5840  017224  000261                          SEC                     ;SCOPE SYNC - SET "C"
5841
5842  017226  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5843
5844  017230  103001                          BCC     4$              ;BR IF "C" CLEAR - IT SHOULD BE
5845
5846  017232  104001                  3$:     ERROR   1               ;RORB FAILED TO CLEAR "C"
5847
5848  017234  020415                  4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5849  017236  001402                          BEQ     TST261          ;;BR IF YES
5850
5851  017240  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5852  017242  104001                  5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
```

```
5853
5854                                    ;;*****************************************************************
5855                                    ;*TEST 261      ASRB TEST - DM2 - ODD ADDRESS
5856                                    ;;*****************************************************************
5857  017244                            TST261:
5858  017244  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5859  017246  012700  000261                   MOV     #261,R0         ;;LOAD R0 WITH TEST NUMBER
5860  017252  013701  017302                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5861  017256  012702  063313                   MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5862  017262  012704  000377                   MOV     #377,R4         ;RESULT S / B = 377
5863  017266  012705  063312                   MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5864  017272  010203                           MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5865  017274  012715  000777                   MOV     #777,(R5)       ;[DEST] = 777
5866  017300  000257                           CCC                     ;SCOPE SYNC "C" = 0
5867
5868  017302  106223                    2$:    ASRB    (R3)+           ;TEST THE ASRB
5869
5870  017304  103401                           BCS     4$              ;BR IF CARRY SET - IT SHOULD BE
5871
5872  017306  104001                    3$:    ERROR   1               ;ASRB FAILED TO SET THE CARRY
5873
5874  017310  022703  063314            4$:    CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5875  017314  001401                           BEQ     6$              ;BR IF YES
5876
5877  017316  104005                    5$:    ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5878
5879  017320  020415                    6$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5880  017322  001402                           BEQ     TST262          ;;BR IF YES
5881
5882  017324  011503                           MOV     (R5),R3         ;GET THE WAS DATA
5883  017326  104001                    7$:    ERROR   1               ;ASRB DELIVERED WRONG RESULT
5884
5885                                    ;;*****************************************************************
5886                                    ;*TEST 262      ASRB TEST - DM1 - ODD ADDRESS
5887                                    ;;*****************************************************************
5888  017330                            TST262:
5889  017330  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5890  017332  012700  000262                   MOV     #262,R0         ;;LOAD R0 WITH TEST NUMBER
5891  017336  013701  017366                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5892  017342  012702  063313                   MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5893  017346  012704  140377                   MOV     #140377,R4      ;RESULT S / B = 140377
5894  017352  012705  063312                   MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5895  017356  010203                           MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5896  017360  012715  100377                   MOV     #100377,(R5)    ;[DEST] = 100377
5897  017364  000261                           SEC                     ;SCOPE SYNC - "C" = 1
5898
5899  017366  106213                    2$:    ASRB    (R3)            ;TEST THE ASRB
5900
5901  017370  103001                           BCC     4$              ;BR IF CARRY CLEAR - IT SHOULD BE
5902
5903  017372  104001                    3$:    ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5904
5905  017374  020415                    4$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5906  017376  001402                           BEQ     TST263          ;;BR IF YES
5907
5908  017400  011503                           MOV     (R5),R3         ;GET THE WAS DATA
```

```
5909  017402  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5910
5911                                  ;;*****************************************************************
5912                                  ;*TEST 263      ASRB TEST - DM2 - EVEN ADDRESS
5913                                  ;;*****************************************************************
5914  017404                          TST263:
5915  017404  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5916  017406  012700  000263                 MOV     #263,R0         ;;LOAD R0 WITH TEST NUMBER
5917  017412  013701  017436                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5918  017416  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5919  017422  012704  000077                 MOV     #77,R4          ;RESULT S / B = 77
5920  017426  010203                         MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5921  017430  012712  000177                 MOV     #177,(R2)       ;[DEST] = 177
5922  017434  000257                         CCC                     ;SCOPE SYNC - "C" = 0
5923
5924  017436  106223                  2$:     ASRB    (R3)+           ;TEST THE ASRB
5925
5926  017440  103401                          BCS     4$              ;BR IF "C" = 1 - IT SHOULD BE
5927
5928  017442  104001                  3$:     ERROR   1               ;ASRB FAILED TO SET "C"
5929
5930  017444  022703  063313          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5931  017450  001401                          BEQ     6$              ;BR IF YES
5932
5933  017452  104005                  5$:     ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5934
5935  017454  020412                  6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5936  017456  001402                          BEQ     TST264          ;;BR IF YES
5937
5938  017460  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5939  017462  104001                  7$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5940
5941                                  ;;*****************************************************************
5942                                  ;*TEST 264      ASRB TEST - DM1 - EVEN ADDRESS
5943                                  ;;*****************************************************************
5944  017464                          TST264:
5945  017464  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5946  017466  012700  000264                 MOV     #264,R0         ;;LOAD R0 WITH TEST NUMBER
5947  017472  013701  017516                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5948  017476  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5949  017502  012704  000303                 MOV     #303,R4         ;RESULT S / B = 303
5950  017506  010203                         MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5951  017510  012712  000206                 MOV     #206,(R2)       ;[DEST] = 206
5952  017514  000261                         SEC                     ;SCOPE SYNC - "C" = 1
5953
5954  017516  106213                  2$:     ASRB    (R3)            ;TEST THE CLRASRB
5955
5956  017520  103001                          BCC     4$              ;BR IF CARRY CLEAR - ITSHOULD BE
5957
5958  017522  104001                  3$:     ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5959
5960  017524  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5961  017526  001402                          BEQ     TST265          ;;BR IF YES
5962
5963  017530  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5964  017532  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
```

```
5965
5966                                    ;;*****************************************************************
5967                                    ;*TEST 265      TST DMO TEST - N:C = 1011
5968                                    ;;*****************************************************************
5969  017534                            TST265:
5970  017534  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5971  017536  012700  000265                    MOV     #265,R0         ;;LOAD R0 WITH TEST NUMBER
5972  017542  013701  017556                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5973  017546  005004                            CLR     R4              ;RESULT S / B = 000000
5974  017550  005003                            CLR     R3              ;[DEST] = 000000
5975  017552  000257                            CCC                     ;CLEAR CODES
5976  017554  000273                            273                     ;N:C=1011
5977
5978  017556  005703                    2$:     TST     R3              ;TEST THE TST
5979
5980  017560  100403                            BMI     3$              ;N:C = 0100 ?
5981  017562  001002                            BNE     3$
5982  017564  102401                            BVS     3$
5983  017566  103001                            BCC     4$
5984
5985  017570  104002                    3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY
5986
5987  017572  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5988  017574  001401                            BEQ     TST266          ;;BR IF YES
5989
5990  017576  104002                    5$:     ERROR   2               ;TST ALTERED THE [DEST]
5991
5992                                    ;;*****************************************************************
5993                                    ;*TEST 266      TST DMO TEST - N:C = 0100
5994                                    ;;*****************************************************************
5995  017600                            TST266:
5996  017600  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5997  017602  012700  000266                    MOV     #266,R0         ;;LOAD R0 WITH TEST NUMBER
5998  017606  013701  017624                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5999  017612  005004                            CLR     R4
6000  017614  005104                            COM     R4              ;RESULT S / B = 177777
6001  017616  010403                            MOV     R4,R3           ;[DEST] = 177777
6002  017620  000257                            CCC                     ;CLEAR CODES
6003  017622  000264                            264                     ;N:C=0100
6004
6005  017624  005703                    2$:     TST     R3              ;TEST THE TST
6006
6007  017626  100003                            BPL     3$              ;N:C = 1000 ?
6008  017630  001402                            BEQ     3$
6009  017632  102401                            BVS     3$
6010  017634  103001                            BCC     4$
6011
6012  017636  104002                    3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY
6013
6014  017640  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
6015  017642  001401                            BEQ     TST267          ;;BR IF YES
6016
6017  017644  104002                    5$:     ERROR   2               ;TST ALTERED THE [DEST]
6018
6019                                    ;;*****************************************************************
6020                                    ;*TEST 267      CLR DMO TEST - N:C = 1011
```

```
6021                                    ;;*****************************************************************
6022  017646                            TST267:
6023  017646  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6024  017650  012700  000267                    MOV     #267,R0         ;;LOAD R0 WITH TEST NUMBER
6025  017654  013701  017672                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6026  017660  005004                            CLR     R4              ;RESULT S / B = 000000
6027  017662  012703  177777                    MOV     #-1,R3          ;[DEST] = 177777
6028  017666  000257                            CCC                     ;CLEAR CODES
6029  017670  000273                            273                     ;N:C = 1011
6030
6031  017672  005003                    2$:     CLR     R3              ;TEST THE CLR
6032
6033  017674  100403                            BMI     3$              ;N:C = 0100 ?
6034  017676  001002                            BNE     3$
6035  017700  102401                            BVS     3$
6036  017702  103001                            BCC     4$
6037
6038  017704  104002                    3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6039
6040  017706  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
6041  017710  001401                            BEQ     TST270          ;;BR IF YES
6042
6043  017712  104002                    5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6044
6045                                    ;;*****************************************************************
6046                                    ;*TEST 270      CLR DMO TEST - N:C = 0000
6047                                    ;;*****************************************************************
6048  017714                            TST270:
6049  017714  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6050  017716  012700  000270                    MOV     #270,R0         ;;LOAD R0 WITH TEST NUMBER
6051  017722  013701  017736                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6052  017726  005004                            CLR     R4              ;RESULT S / B = 000000
6053  017730  012703  177777                    MOV     #-1,R3          ;[DEST] = 177777
6054  017734  000257                            CCC                     ;CLEAR CODES
6055
6056  017736  005003                    2$:     CLR     R3              ;TEST THE CLR
6057
6058  017740  100403                            BMI     3$              ;N:C = 0100 ?
6059  017742  001002                            BNE     3$
6060  017744  102401                            BVS     3$
6061  017746  103001                            BCC     4$
6062
6063  017750  104002                    3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6064
6065  017752  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
6066  017754  001401                            BEQ     TST271          ;;BR IF YES
6067
6068  017756  104002                    5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6069
6070                                    ;;*****************************************************************
6071                                    ;*TEST 271      COM DMO TEST - N:C = 0110
6072                                    ;;*****************************************************************
6073  017760                            TST271:
6074  017760  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6075  017762  012700  000271                    MOV     #271,R0         ;;LOAD R0 WITH TEST NUMBER
6076  017766  013701  020006                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
6077  017772  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6078  017776  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
6079  020002  000257                  CCC                     ;CLEAR CODES
6080  020004  000266                  266                     ;N:C = 0110
6081
6082  020006  005103          2$:     COM     R3              ;TEST THE COM
6083
6084  020010  100003                  BPL     3$              ;N:C = 1001 ?
6085  020012  001402                  BEQ     3$
6086  020014  102401                  BVS     3$
6087  020016  103401                  BCS     4$
6088
6089  020020  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6090
6091  020022  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6092  020024  001401                  BEQ     TST272          ;;BR IF YES
6093
6094  020026  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6095
6096                          ;;**************************************************************
6097                          ;*TEST 272       COM DMO TEST - N:C = 1001
6098                          ;;**************************************************************
6099  020030                  TST272:
6100  020030  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6101  020032  012700  000272          MOV     #272,R0         ;;LOAD R0 WITH TEST NUMBER
6102  020036  013701  020054          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6103  020042  005004                  CLR     R4              ;RESULT S / B = 000000
6104  020044  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6105  020050  000257                  CCC                     ;CLEAR CODES
6106  020052  000271                  271                     ;N:C = 1001
6107
6108  020054  005103          2$:     COM     R3              ;TEST THE COM
6109
6110  020056  100403                  BMI     3$              ;N:C = 0101 ?
6111  020060  001002                  BNE     3$
6112  020062  102401                  BVS     3$
6113  020064  103401                  BCS     4$
6114
6115  020066  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6116
6117  020070  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6118  020072  001401                  BEQ     TST273          ;;BR IF YES
6119
6120  020074  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6121
6122                          ;;**************************************************************
6123                          ;*TEST 273       INC DMO TEST - N:C = 1011
6124                          ;;**************************************************************
6125  020076                  TST273:
6126  020076  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6127  020100  012700  000273          MOV     #273,R0         ;;LOAD R0 WITH TEST NUMBER
6128  020104  013701  020122          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6129  020110  005004                  CLR     R4              ;RESULT S / B = 000000
6130  020112  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6131  020116  000257                  CCC                     ;CLEAR CODES
6132  020120  000273                  273                     ;N:C = 1011
```

```
6133
6134  020122  005203                  2$:     INC     R3              ;TEST THE INC
6135
6136  020124  100403                          BMI     3$              ;N:C = 0101 ?
6137  020126  001002                          BNE     3$
6138  020130  102401                          BVS     3$
6139  020132  103401                          BCS     4$
6140
6141  020134  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
6142
6143  020136  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6144  020140  001401                          BEQ     TST274          ;;BR IF YES
6145
6146  020142  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
6147
6148                                  ;;**********************************************************************
6149                                  ;*TEST 274       INC DMO TEST - N:C = 0100
6150                                  ;;**********************************************************************
6151  020144                          TST274:
6152  020144  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6153  020146  012700  000274                  MOV     #274,R0         ;;LOAD R0 WITH TEST NUMBER
6154  020152  013701  020172                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6155  020156  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6156  020162  012703  077777                  MOV     #77777,R3       ;[DEST] = 77777
6157  020166  000257                          CCC                     ;CLEAR CODES
6158  020170  000264                          264                     ;N:C = 0100
6159
6160  020172  005203                  2$:     INC     R3              ;TEST THE INC
6161
6162  020174  100003                          BPL     3$              ;N:C = 1010 ?
6163  020176  001402                          BEQ     3$
6164  020200  102001                          BVC     3$
6165  020202  103001                          BCC     4$
6166
6167  020204  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
6168
6169  020206  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6170  020210  001401                          BEQ     TST275          ;;BR IF YES
6171
6172  020212  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
6173
6174                                  ;;**********************************************************************
6175                                  ;*TEST 275       DEC DMO TEST - N:C = 1011
6176                                  ;;**********************************************************************
6177  020214                          TST275:
6178  020214  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6179  020216  012700  000275                  MOV     #275,R0         ;;LOAD R0 WITH TEST NUMBER
6180  020222  013701  020240                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6181  020226  005004                          CLR     R4              ;RESULT S / B = 000000
6182  020230  012703  000001                  MOV     #1,R3           ;[DEST] = 1
6183  020234  000257                          CCC                     ;CLEAR CODES
6184  020236  000273                          273                     ;N:C = 1011
6185
6186  020240  005303                  2$:     DEC     R3              ;TEST THE DEC
6187
6188  020242  100403                          BMI     3$              ;N:C = 0101 ?
```

```
6189  020244  001002                          BNE     3$
6190  020246  102401                          BVS     3$
6191  020250  103401                          BCS     4$
6192
6193  020252  104002                  3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
6194
6195  020254  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6196  020256  001401                          BEQ     TST276          ;;BR IF YES
6197
6198  020260  104002                  5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
6199
6200                                  ;;**********************************************************************
6201                                  ;*TEST 276       DEC DMD TEST - N:C = 1100
6202                                  ;;**********************************************************************
6203  020262                          TST276:
6204  020262  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6205  020264  012700  000276                  MOV     #276,R0         ;;LOAD R0 WITH TEST NUMBER
6206  020270  013701  020310                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6207  020274  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
6208  020300  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
6209  020304  000257                          CCC                     ;CLEAR CODES
6210  020306  000274                          274                     ;N:C = 1100
6211
6212  020310  005303                  2$:     DEC     R3              ;TEST THE DEC
6213
6214  020312  100403                          BMI     3$              ;N:C = 0010 ?
6215  020314  001402                          BEQ     3$
6216  020316  102001                          BVC     3$
6217  020320  103001                          BCC     4$
6218
6219  020322  104002                  3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
6220
6221  020324  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6222  020326  001401                          BEQ     TST277          ;;BR IF YES
6223
6224  020330  104002                  5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
6225
6226                                  ;;**********************************************************************
6227                                  ;*TEST 277       DEC DMD TEST - N:C = 0000
6228                                  ;;**********************************************************************
6229  020332                          TST277:
6230  020332  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6231  020334  012700  000277                  MOV     #277,R0         ;;LOAD R0 WITH TEST NUMBER
6232  020340  013701  020354                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6233  020344  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
6234  020350  005003                          CLR     R3              ;[DEST] = 000000
6235  020352  000257                          CCC                     ;CLEAR CODES
6236
6237  020354  005303                  2$:     DEC     R3              ;TEST THE DEC
6238
6239  020356  100003                          BPL     3$              ;N:C = 1000 ?
6240  020360  001402                          BEQ     3$
6241  020362  102401                          BVS     3$
6242  020364  103001                          BCC     4$
6243
6244  020366  104002                  3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
```

```
6245
6246  020370  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6247  020372  001401                          BEQ     TST300          ;;BR IF YES
6248
6249  020374  104002                  5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
6250
6251                                  ;;**************************************************************
6252                                  ;*TEST 300       ASL DMO TEST - N:C = 1000
6253                                  ;;**************************************************************
6254  020376                          TST300:
6255  020376  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6256  020400  012700  000300                  MOV     #300,R0         ;;LOAD R0 WITH TEST NUMBER
6257  020404  013701  020422                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6258  020410  005004                          CLR     R4              ;RESULT S / B = 000000
6259  020412  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
6260  020416  000257                          CCC                     ;CLEAR CODES
6261  020420  000270                          SEN                     ;N:C = 1000
6262
6263  020422  006303                  2$:     ASL     R3              ;TEST THE ASL
6264
6265  020424  100403                          BMI     3$              ;N:C = 0111 ?
6266  020426  001002                          BNE     3$
6267  020430  102001                          BVC     3$
6268  020432  103401                          BCS     4$
6269
6270  020434  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6271
6272  020436  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6273  020440  001401                          BEQ     TST301          ;;BR IF YES
6274
6275  020442  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6276
6277                                  ;;**************************************************************
6278                                  ;*TEST 301       ASL DMO TEST - N:C = 0101
6279                                  ;;**************************************************************
6280  020444                          TST301:
6281  020444  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6282  020446  012700  000301                  MOV     #301,R0         ;;LOAD R0 WITH TEST NUMBER
6283  020452  013701  020472                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6284  020456  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6285  020462  012703  040000                  MOV     #40000,R3       ;[DEST] = 40000
6286  020466  000257                          CCC                     ;CLEAR CODES
6287  020470  000265                          265                     ;N:C = 0101
6288
6289  020472  006303                  2$:     ASL     R3              ;TEST THE ASL
6290
6291  020474  100003                          BPL     3$              ;N:C = 1010 ?
6292  020476  001402                          BEQ     3$
6293  020500  102001                          BVC     3$
6294  020502  103001                          BCC     4$
6295
6296  020504  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6297
6298  020506  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6299  020510  001401                          BEQ     TST302          ;;BR IF YES
6300
```

```
6301  020512  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6302
6303                                  ;;*****************************************************************
6304                                  ;*TEST 302       ASL DMO TEST - N:C = 0010
6305                                  ;;*****************************************************************
6306  020514                          TST302:
6307  020514  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6308  020516  012700  000302                  MOV     #302,R0         ;;LOAD R0 WITH TEST NUMBER
6309  020522  013701  020536                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6310  020526  005004                          CLR     R4              ;RESULT S  /  B = 000000
6311  020530  005003                          CLR     R3              ;[DEST] = 000000
6312  020532  000257                          CCC                     ;CLEAR CODES
6313  020534  000262                          SEV                     ;N:C = 0010
6314
6315  020536  006303                  2$:     ASL     R3              ;TEST THE ASL
6316
6317  020540  100403                          BMI     3$              ;N:C = 0100 ?
6318  020542  001002                          BNE     3$
6319  020544  102401                          BVS     3$
6320  020546  103001                          BCC     4$
6321
6322  020550  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6323
6324  020552  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6325  020554  001401                          BEQ     TST303          ;;BR IF YES
6326
6327  020556  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6328
6329                                  ;;*****************************************************************
6330                                  ;*TEST 303       ROL DMO TEST - N:C = 1101
6331                                  ;;*****************************************************************
6332  020560                          TST303:
6333  020560  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6334  020562  012700  000303                  MOV     #303,R0         ;;LOAD R0 WITH TEST NUMBER
6335  020566  013701  020606                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6336  020572  012704  052525                  MOV     #52525,R4       ;RESULT S  /  B = 52525
6337  020576  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
6338  020602  000257                          CCC                     ;CLEAR CODES
6339  020604  000275                          275                     ;N:C * 1101
6340
6341  020606  006103                  2$:     ROL     R3              ;TEST THE ROL
6342
6343  020610  100403                          BMI     3$              ;N:C = 0011 ?
6344  020612  001402                          BEQ     3$
6345  020614  102001                          BVC     3$
6346  020616  103401                          BCS     4$
6347
6348  020620  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6349
6350  020622  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6351  020624  001401                          BEQ     TST304          ;;BR IF YES
6352
6353  020626  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6354
6355                                  ;;*****************************************************************
6356                                  ;*TEST 304       ROL DMO TEST - N:C = 0101
```

```
6357                                    ;;*****************************************************************
6358  020630                            T$T304:
6359  020630  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6360  020632  012700  000304                    MOV     #304,R0         ;;LOAD R0 WITH TEST NUMBER
6361  020636  013701  020656                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6362  020642  012704  125253                    MOV     #125253,R4      ;RESULT S / B = 125253
6363  020646  012703  052525                    MOV     #52525,R3       ;[DEST] = 52525
6364  020652  000257                            CCC                     ;CLEAR CODES
6365  020654  000265                            265                     ;N:C = 0101
6366
6367  020656  006103                    2$:     ROL     R3              ;TEST THE ROL
6368
6369  020660  100003                            BPL     3$              ;N:C = 1010 ?
6370  020662  001402                            BEQ     3$
6371  020664  102001                            BVC     3$
6372  020666  103001                            BCC     4$
6373
6374  020670  104002                    3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6375  020672  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
6376  020674  001401                            BEQ     T$T305          ;;BR IF YES
6377
6378  020676  104002                    5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6379
6380                                    ;;*****************************************************************
6381                                    ;*TEST 305      ROL DMD TEST - N:C = 0010
6382                                    ;;*****************************************************************
6383  020700                            T$T305:
6384  020700  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6385  020702  012700  000305                    MOV     #305,R0         ;;LOAD R0 WITH TEST NUMBER
6386  020706  013701  020722                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6387  020712  005004                            CLR     R4              ;RESULT S / B = 000000
6388  020714  005003                            CLR     R3              ;[DEST] = 000000
6389  020716  000257                            CCC                     ;CLEAR CODES
6390  020720  000262                            SEV                     ;N:C = 0010
6391
6392  020722  006103                    2$:     ROL     R3              ;TEST THE ROL
6393
6394  020724  100403                            BMI     3$              ;N:C = 0100 ?
6395  020726  001002                            BNE     3$
6396  020730  102401                            BVS     3$
6397  020732  103001                            BCC     4$
6398
6399  020734  104002                    3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6400
6401  020736  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
6402  020740  001401                            BEQ     T$T306          ;;BR IF YES
6403
6404  020742  104002                    5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
6405
6406                                    ;;*****************************************************************
6407                                    ;*TEST 306      ADC DMD TEST - N:C = 0101
6408                                    ;;*****************************************************************
6409  020744                            T$T306:
6410  020744  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6411  020746  012700  000306                    MOV     #306,R0         ;;LOAD R0 WITH TEST NUMBER
6412  020752  013701  020772                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
6413  020756  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
6414  020762  012703  077777          MOV     #77777,R3       ;[DEST] = 77777
6415  020766  000257                  CCC                     ;CLEAR CODES
6416  020770  000265                  265                     ;N:C = 0101
6417
6418  020772  005503          2$:     ADC     R3              ;TEST THE ADC
6419
6420  020774  100003                  BPL     3$              ;N:C = 1010 ?
6421  020776  001402                  BEQ     3$
6422  021000  102001                  BVC     3$
6423  021002  103001                  BCC     4$
6424
6425  021004  104002          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
6426
6427  021006  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6428  021010  001401                  BEQ     TST307          ;;BR IF YES
6429
6430  021012  104002          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
6431
6432                          ;;**********************************************************************
6433                          ;*TEST 307       ADC DMO TEST - N:C = 1011
6434                          ;;**********************************************************************
6435  021014                  TST307:
6436  021014  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6437  021016  012700  000307          MOV     #307,R0         ;;LOAD R0 WITH TEST NUMBER
6438  021022  013701  021040          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6439  021026  005004                  CLR     R4              ;RESULT S / B = 000000
6440  021030  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6441  021034  000257                  CCC                     ;CLEAR CODES
6442  021036  000273                  273                     ;N:C = 1011
6443
6444  021040  005503          2$:     ADC     R3              ;TEST THE ADC
6445
6446  021042  100403                  BMI     3$              ;N:C = 0101 ?
6447  021044  001002                  BNE     3$
6448  021046  102401                  BVS     3$
6449  021050  103401                  BCS     4$
6450
6451  021052  104002          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
6452
6453  021054  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6454  021056  001401                  BEQ     TST310          ;;BR IF YES
6455
6456  021060  104002          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
6457
6458                          ;;**********************************************************************
6459                          ;*TEST 310       ADC DMO TEST - N:C = 1010
6460                          ;;**********************************************************************
6461  021062                  TST310:
6462  021062  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6463  021064  012700  000310          MOV     #310,R0         ;;LOAD R0 WITH TEST NUMBER
6464  021070  013701  021110          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6465  021074  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
6466  021100  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6467  021104  000257                  CCC                     ;CLEAR CODES
6468  021106  000272                  272                     ;N:C = 1010
```

```
6469
6470  021110  005503                  2$:     ADC     R3              ;TEST THE ADC
6471
6472  021112  100003                          BPL     3$              ;N:C = 1000 ?
6473  021114  001402                          BEQ     3$
6474  021116  102401                          BVS     3$
6475  021120  103001                          BCC     4$
6476
6477  021122  104002                  3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
6478
6479  021124  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6480  021126  001401                          BEQ     TST311          ;;BR IF YES
6481
6482  021130  104002                  5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
6483
6484                                  ;;**********************************************************************
6485                                  ;*TEST 311       SBC DMO TEST - N:C = 1011
6486                                  ;;**********************************************************************
6487  021132                          TST311:
6488  021132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6489  021134  012700  000311                  MOV     #311,R0         ;;LOAD R0 WITH TEST NUMBER
6490  021140  013701  021156                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6491  021144  005004                          CLR     R4              ;RESULT S / B = 000000
6492  021146  012703  000001                  MOV     #1,R3           ;[DEST] = +1
6493  021152  000257                          CCC                     ;CLEAR CODES
6494  021154  000273                          273                     ;N:C = 1011
6495
6496  021156  005603                  2$:     SBC     R3              ;TEST THE SBC
6497
6498  021160  100403                          BMI     3$              ;N:C = 0100 ?
6499  021162  001002                          BNE     3$
6500  021164  102401                          BVS     3$
6501  021166  103001                          BCC     4$
6502
6503  021170  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6504
6505  021172  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6506  021174  001401                          BEQ     TST312          ;;BR IF YES
6507
6508  021176  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6509
6510                                  ;;**********************************************************************
6511                                  ;*TEST 312       SBC DMO TEST - N:C = 0101
6512                                  ;;**********************************************************************
6513  021200                          TST312:
6514  021200  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6515  021202  012700  000312                  MOV     #312,R0         ;;LOAD R0 WITH TEST NUMBER
6516  021206  013701  021226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6517  021212  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
6518  021216  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
6519  021222  000257                          CCC                     ;CLEAR CODES
6520  021224  000265                          265                     ;N:C = 0101
6521
6522  021226  005603                  2$:     SBC     R3              ;TEST THE SBC
6523
6524  021230  100403                          BMI     3$              ;N:C = 0010 ?
```

```
6525  021232  001402                          BEQ     3$
6526  021234  102001                          BVC     3$
6527  021236  103001                          BCC     4$
6528
6529  021240  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6530
6531  021242  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6532  021244  001401                          BEQ     TST313          ;;BR IF YES
6533
6534  021246  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6535
6536                                  ;;**********************************************************************
6537                                  ;*TEST 313       SBC DMO TEST - N:C = 1110
6538                                  ;;**********************************************************************
6539  021250                          TST313:
6540  021250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6541  021252  012700  000313                  MOV     #313,R0         ;;LOAD R0 WITH TEST NUMBER
6542  021256  013701  021276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6543  021262  012704  000001                  MOV     #1,R4           ;RESULT S / B = 1
6544  021266  012703  000001                  MOV     #1,R3           ;[DEST] = 1
6545  021272  000257                          CCC                     ;CLEAR CODES
6546  021274  000276                          276                     ;N:C = 1110
6547
6548  021276  005603                  2$:     SBC     R3              ;TEST THE SBC
6549
6550  021300  100403                          BMI     3$              ;N:C = 0000 ?
6551  021302  001402                          BEQ     3$
6552  021304  102401                          BVS     3$
6553  021306  103001                          BCC     4$
6554
6555  021310  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6556
6557  021312  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6558  021314  001401                          BEQ     TST314          ;;BR IF YES
6559
6560  021316  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6561
6562                                  ;;**********************************************************************
6563                                  ;*TEST 314       SBC DMO TEST - N:C = 0111
6564                                  ;;**********************************************************************
6565  021320                          TST314:
6566  021320  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6567  021322  012700  000314                  MOV     #314,R0         ;;LOAD R0 WITH TEST NUMBER
6568  021326  013701  021344                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6569  021332  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
6570  021336  005003                          CLR     R3              ;[DEST] = 000000
6571  021340  000257                          CCC                     ;CLEAR CODES
6572  021342  000267                          267                     ;N:C = 0111
6573
6574  021344  005603                  2$:     SBC     R3              ;TEST THE SBC
6575
6576  021346  100003                          BPL     3$              ;N:C = 1001 ?
6577  021350  001402                          BEQ     3$
6578  021352  102401                          BVS     3$
6579  021354  103401                          BCS     4$
6580
```

```
6581  021356  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6582
6583  021360  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6584  021362  001401                          BEQ     TST315          ;;BR IF YES
6585
6586  021364  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6587
6588                                  ;;**********************************************************************
6589                                  ;*TEST 315       TST DM1 TEST - N:C = 1011
6590                                  ;;**********************************************************************
6591  021366                          TST315:
6592  021366  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6593  021370  012700  000315                  MOV     #315,R0         ;;LOAD R0 WITH TEST NUMBER
6594  021374  013701  021414                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6595  021400  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6596  021404  005004                          CLR     R4              ;RESULT S / B = 000000
6597  021406  005012                          CLR     (R2)            ;[DEST] = 000000
6598  021410  000257                          CCC                     ;CLEAR CODES
6599  021412  000273                          273                     ;N:C=1011
6600
6601  021414  005712                  2$:     TST     (R2)            ;TEST THE TST
6602
6603  021416  100403                          BMI     3$              ;N:C = 0100 ?
6604  021420  001002                          BNE     3$
6605  021422  102401                          BVS     3$
6606  021424  103001                          BCC     4$
6607
6608  021426  104001                  3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
6609
6610  021430  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6611  021432  001402                          BEQ     TST316          ;;BR IF YES
6612
6613  021434  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6614  021436  104001                  5$:     ERROR   1               ;TST ALTERED THE [DEST]
6615
6616                                  ;;**********************************************************************
6617                                  ;*TEST 316       TST DM1 TEST - N:C = 0100
6618                                  ;;**********************************************************************
6619  021440                          TST316:
6620  021440  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6621  021442  012700  000316                  MOV     #316,R0         ;;LOAD R0 WITH TEST NUMBER
6622  021446  013701  021472                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6623  021452  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6624  021456  005004                          CLR     R4
6625  021460  005104                          COM     R4              ;RESULT S / B = 177777
6626  021462  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
6627  021466  000257                          CCC                     ;CLEAR CODES
6628  021470  000264                          264                     ;N:C=0100
6629
6630  021472  005712                  2$:     TST     (R2)            ;TEST THE TST
6631
6632  021474  100003                          BPL     3$              ;N:C = 1000 ?
6633  021476  001402                          BEQ     3$
6634  021500  102401                          BVS     3$
6635  021502  103001                          BCC     4$
6636
```

```
6637  021504  104001                    3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
6638
6639  021506  020412                    4$:     CMP     R4,(R2)         ;RESULT OK ?
6640  021510  001402                            BEQ     TST317          ;;BR IF YES
6641
6642  021512  011203                            MOV     (R2),R3         ;GET THE WAS DATA
6643  021514  104001                    5$:     ERROR   1               ;TST ALTERED THE [DEST]
6644
6645                                    ;;**********************************************************************
6646                                    ;*TEST 317      CLR DM1 TEST - N:C = 1011
6647                                    ;;**********************************************************************
6648  021516                            TST317:
6649  021516  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6650  021520  012700  000317                    MOV     #317,R0         ;;LOAD R0 WITH TEST NUMBER
6651  021524  013701  021560                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6652                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT6
6653  021530  032737  000100  063234            BIT     #BIT6,@#BPTLOC  ;BREAKPOINT HALT SET ??
6654  021536  001401                            BEQ     .+4             ;BR IF NOT
6655  021540  000000                            HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
6656  021542  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6657  021546  005004                            CLR     R4              ;RESULT S / B = 000000
6658  021550  012712  177777                    MOV     #-1,(R2)        ;[DEST] = 177777
6659  021554  000257                            CCC                     ;CLEAR CODES
6660  021556  000273                            273                     ;N:C = 1011
6661
6662  021560  005012                    2$:     CLR     (R2)            ;TEST THE CLR
6663
6664  021562  100403                            BMI     3$              ;N:C = 0100 ?
6665  021564  001002                            BNE     3$
6666  021566  102401                            BVS     3$
6667  021570  103001                            BCC     4$
6668
6669  021572  104001                    3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
6670
6671  021574  020412                    4$:     CMP     R4,(R2)         ;RESULT OK ?
6672  021576  001402                            BEQ     TST320          ;;BR IF YES
6673
6674  021600  011203                            MOV     (R2),R3         ;GET THE WAS DATA
6675  021602  104001                    5$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
6676
6677                                    ;;**********************************************************************
6678                                    ;*TEST 320      CLR DM2 TEST - N:C = 0000
6679                                    ;;**********************************************************************
6680  021604                            TST320:
6681  021604  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6682  021606  012700  000320                    MOV     #320,R0         ;;LOAD R0 WITH TEST NUMBER
6683  021612  013701  021632                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6684  021616  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6685  021622  005004                            CLR     R4              ;RESULT S / B = 000000
6686  021624  013712  063324                    MOV     @#DWTA+2,(R2)   ;[DEST] = 177777
6687  021630  000257                            CCC                     ;CLEAR CODES
6688
6689  021632  005022                    2$:     CLR     (R2)+           ;TEST THE CLR
6690
6691  021634  100403                            BMI     3$              ;N:C = 0100 ?
6692  021636  001002                            BNE     3$
```

```
021640  102401                      BVS     3$
021642  103001                      BCC     4$

021644  104001          3$:         ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY

021646  022702  063314  4$:         CMP     #MBUF0+2,R2     ;DID CLR INCREMENT DEST REG
021652  001401                      BEQ     6$              ;BR IF YES

021654  104005          5$:         ERROR   5               ;CLR FAILED TO UPDATE DEST REG

021656  020442          6$:         CMP     R4,-(R2)                ;RESULT OK ?
021660  001402                      BEQ     TST321          ;;BR IF YES

021662  011203                      MOV     (R2),R3         ;GET THE WAS DATA
021664  104001          7$:         ERROR   1               ;CLR DELIVERED THE WRONG RESULT

                        ;;**********************************************************************
                        ;*TEST 321      COM DM1 TEST - N:C = 0110
                        ;;**********************************************************************
021666                  TST321:
021666  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
021670  012700  000321              MOV     #321,R0         ;;LOAD R0 WITH TEST NUMBER
021674  013701  021720              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
021700  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
021704  012704  125252              MOV     #125252,R4      ;RESULT S / B = 125252
021710  012712  052525              MOV     #52525,(R2)     ;[DEST] = 52525
021714  000257                      CCC                     ;CLEAR CODES
021716  000266                      266                     ;N:C = 0110

021720  005112          2$:         COM     (R2)            ;TEST THE CLR

021722  100003                      BPL     3$              ;N:C = 1001 ?
021724  001402                      BEQ     3$
021726  102401                      BVS     3$
021730  103401                      BCS     4$

021732  104001          3$:         ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
021734  020412          4$:         CMP     R4,(R2)         ;RESULT OK ?
021736  001402                      BEQ     TST322          ;;BR IF YES

021740  011203                      MOV     (R2),R3         ;GET THE WAS DATA
021742  104001          5$:         ERROR   1               ;COM DELIVERED THE WRONG RESULT

                        ;;**********************************************************************
                        ;*TEST 322      COM DM1 TEST - N:C = 1001
                        ;;**********************************************************************
021744                  TST322:
021744  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
021746  012700  000322              MOV     #322,R0         ;;LOAD R0 WITH TEST NUMBER
021752  013701  021774              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
021756  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
021762  005004                      CLR     R4              ;RESULT S / B = 000000
021764  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
021770  000257                      CCC                     ;CLEAR CODES
021772  000271                      271                     ;N:C = 1001
```

```
6749  021774  005112                  2$:     COM     (R2)            ;TEST THE COM
6750
6751  021776  100403                          BMI     3$              ;N:C = 0101 ?
6752  022000  001002                          BNE     3$
6753  022002  102401                          BVS     3$
6754  022004  103401                          BCS     4$
6755
6756  022006  104001                  3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6757  022010  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6758  022012  001402                          BEQ     TST323          ;;BR IF YES
6759
6760  022014  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6761  022016  104001                  5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6762
6763                                  ;;**********************************************************************
6764                                  ;*TEST 323       INC DM1 TEST - N:C = 1011
6765                                  ;;**********************************************************************
6766  022020                          TST323:
6767  022020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6768  022022  012700  000323                  MOV     #323,R0         ;;LOAD R0 WITH TEST NUMBER
6769  022026  013701  022050                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6770  022032  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6771  022036  005004                          CLR     R4              ;RESULT S / B = 000000
6772  022040  012712  177777                  MOV     #-1,(R2)                ;[DEST] = 177777
6773  022044  000257                          CCC                     ;CLEAR CODES
6774  022046  000273                          273                     ;N:C = 1011
6775
6776  022050  005212                  2$:     INC     (R2)            ;TEST THE INC
6777
6778  022052  100403                          BMI     3$              ;N:C = 0101 ?
6779  022054  001002                          BNE     3$
6780  022056  102401                          BVS     3$
6781  022060  103401                          BCS     4$
6782
6783  022062  104001                  3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6784  022064  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6785  022066  001402                          BEQ     TST324          ;;BR IF YES
6786
6787  022070  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6788  022072  104001                  5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6789
6790                                  ;;**********************************************************************
6791                                  ;*TEST 324       INC DM1 TEST - N:C = 0100
6792                                  ;;**********************************************************************
6793  022074                          TST324:
6794  022074  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6795  022076  012700  000324                  MOV     #324,R0         ;;LOAD R0 WITH TEST NUMBER
6796  022102  013701  022126                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6797  022106  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6798  022112  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6799  022116  012712  077777                  MOV     #77777,(R2)     ;[DEST] = 77777
6800  022122  000257                          CCC                     ;CLEAR CODES
6801  022124  000264                          264                     ;N:C = 0100
6802
6803  022126  005212                  2$:     INC     (R2)            ;TEST THE INC
6804
```

```
6805  022130  100003                  BPL     3$              ;N:C = 1010 ?
6806  022132  001402                  BEQ     3$
6807  022134  102001                  BVC     3$
6808  022136  103001                  BCC     4$
6809
6810  022140  104001          3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6811  022142  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6812  022144  001402                  BEQ     TST325          ;;BR IF YES
6813
6814  022146  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6815  022150  104001          5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6816
6817                          ;;**********************************************************************
6818                          ;*TEST 325      DEC DM1 TEST - N:C = 1011
6819                          ;;**********************************************************************
6820  022152                  TST325:
6821  022152  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6822  022154  012700  000325          MOV     #325,R0         ;;LOAD R0 WITH TEST NUMBER
6823  022160  013701  022202          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6824  022164  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6825  022170  005004                  CLR     R4              ;RESULT S / B = 000000
6826  022172  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
6827  022176  000257                  CCC                     ;CLEAR CODES
6828  022200  000273                  273                     ;N:C = 1011
6829
6830  022202  005312          2$:     DEC     (R2)            ;TEST THE DEC
6831
6832  022204  100403                  BMI     3$              ;N:C = 0101 ?
6833  022206  001002                  BNE     3$
6834  022210  102401                  BVS     3$
6835  022212  103401                  BCS     4$
6836
6837  022214  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6838  022216  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6839  022220  001402                  BEQ     TST326          ;;BR IF YES
6840
6841  022222  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6842  022224  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6843
6844                          ;;**********************************************************************
6845                          ;*TEST 326      DEC DM1 TEST - N:C = 1100
6846                          ;;**********************************************************************
6847  022226                  TST326:
6848  022226  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6849  022230  012700  000326          MOV     #326,R0         ;;LOAD R0 WITH TEST NUMBER
6850  022234  013701  022260          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6851  022240  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6852  022244  012704  077777          MOV     #77777,R4       ;RESULT S / B = 77777
6853  022250  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
6854  022254  000257                  CCC                     ;CLEAR CODES
6855  022256  000274                  274                     ;N:C = 1100
6856
6857  022260  005312          2$:     DEC     (R2)            ;TEST THE DEC
6858
6859  022262  100403                  BMI     3$              ;N:C = 0010 ?
6860  022264  001402                  BEQ     3$
```

```
6861  022266  102001                    BVC     3$
6862  022270  103001                    BCC     4$
6863
6864  022272  104001            3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6865  022274  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
6866  022276  001402                    BEQ     TST327          ;;BR IF YES
6867
6868  022300  011203                    MOV     (R2),R3         ;GET THE WAS DATA
6869  022302  104001            5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6870
6871                            ;;**********************************************************************
6872                            ;*TEST 327      DEC DM1 TEST - N:C = 0000
6873                            ;;**********************************************************************
6874  022304                    TST327:
6875  022304  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6876  022306  012700  000327            MOV     #327,R0         ;;LOAD R0 WITH TEST NUMBER
6877  022312  013701  022332            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6878  022316  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6879  022322  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
6880  022326  005012                    CLR     (R2)            ;[DEST] = 000000
6881  022330  000257                    CCC                     ;CLEAR CODES
6882
6883  022332  005312            2$:     DEC     (R2)            ;TEST THE DEC
6884
6885  022334  100003                    BPL     3$              ;N:C = 1000 ?
6886  022336  001402                    BEQ     3$
6887  022340  102401                    BVS     3$
6888  022342  103001                    BCC     4$
6889
6890  022344  104001            3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6891  022346  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
6892  022350  001402                    BEQ     TST330          ;;BR IF YES
6893
6894  022352  011203                    MOV     (R2),R3         ;GET THE WAS DATA
6895  022354  104001            5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6896
6897                            ;;**********************************************************************
6898                            ;*TEST 330      ASL DM1 TEST - N:C = 1000
6899                            ;;**********************************************************************
6900  022356                    TST330:
6901  022356  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6902  022360  012700  000330            MOV     #330,R0         ;;LOAD R0 WITH TEST NUMBER
6903  022364  013701  022406            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6904  022370  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6905  022374  005004                    CLR     R4              ;RESULT S / B = 000000
6906  022376  012712  100000            MOV     #100000,(R2)    ;[DEST] = 100000
6907  022402  000257                    CCC                     ;CLEAR CODES
6908  022404  000270                    SEN                     ;N:C = 1000
6909
6910  022406  006312            2$:     ASL     (R2)            ;TEST THE ASL
6911
6912  022410  100403                    BMI     3$              ;N:C = 0111 ?
6913  022412  001002                    BNE     3$
6914  022414  102001                    BVC     3$
6915  022416  103401                    BCS     4$
6916
```

```
6917  022420  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6918  022422  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6919  022424  001402                          BEQ     TST331          ;;BR IF YES
6920
6921  022426  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6922  022430  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
6923
6924                                  ;;**********************************************************************
6925                                  ;*TEST 331       ASL DM1 TEST - N:C = 0101
6926                                  ;;**********************************************************************
6927  022432                          TST331:
6928  022432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6929  022434  012700  000331                  MOV     #331,R0         ;;LOAD R0 WITH TEST NUMBER
6930  022440  013701  022464                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6931  022444  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6932  022450  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6933  022454  012712  040000                  MOV     #40000,(R2)     ;(DEST) = 40000
6934  022460  000257                          CCC                     ;CLEAR CODES
6935  022462  000265                          265                     ;N:C = 0101
6936
6937  022464  006312                  2$:     ASL     (R2)            ;TEST THE ASL
6938
6939  022466  100003                          BPL     3$              ;N:C = 1010 ?
6940  022470  001402                          BEQ     3$
6941  022472  102001                          BVC     3$
6942  022474  103001                          BCC     4$
6943
6944  022476  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6945  022500  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6946  022502  001402                          BEQ     TST332          ;;BR IF YES
6947
6948  022504  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6949  022506  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
6950
6951                                  ;;**********************************************************************
6952                                  ;*TEST 332       ASL DM1 TEST - N:C = 0010
6953                                  ;;**********************************************************************
6954  022510                          TST332:
6955  022510  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6956  022512  012700  000332                  MOV     #332,R0         ;;LOAD R0 WITH TEST NUMBER
6957  022516  013701  022536                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6958  022522  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6959  022526  005004                          CLR     R4              ;RESULT S / B = 000000
6960  022530  005012                          CLR     (R2)            ;(DEST) = 000000
6961  022532  000257                          CCC                     ;CLEAR CODES
6962  022534  000262                          SEV                     ;N:C = 0010
6963
6964  022536  006312                  2$:     ASL     (R2)            ;TEST THE ASL
6965
6966  022540  100403                          BMI     3$              ;N:C = 0100 ?
6967  022542  001002                          BNE     3$
6968  022544  102401                          BVS     3$
6969  022546  103001                          BCC     4$
6970
6971  022550  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6972  022552  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
```

```
6973  022554  001402                          BEQ     TST333          ;;BR IF YES
6974
6975  022556  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6976  022560  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
6977
6978                                  ;;**********************************************************************
6979                                  ;*TEST 333       ROL DM1 TEST - N:C = 1101
6980                                  ;;**********************************************************************
6981  022562                          TST333:
6982  022562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6983  022564  012700  000333                  MOV     #333,R0         ;;LOAD R0 WITH TEST NUMBER
6984  022570  013701  022614                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6985  022574  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6986  022600  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
6987  022604  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
6988  022610  000257                          CCC                     ;CLEAR CODES
6989  022612  000275                          275                     ;N:C = 1101
6990
6991  022614  006112                  2$:     ROL     (R2)            ;TEST THE ROL
6992
6993  022616  100403                          BMI     3$              ;N:C = 0011 ?
6994  022620  001402                          BEQ     3$
6995  022622  102001                          BVC     3$
6996  022624  103401                          BCS     4$
6997
6998  022626  104001                  3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
6999  022630  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7000  022632  001402                          BEQ     TST334          ;;BR IF YES
7001
7002  022634  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7003  022636  104001                  5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7004
7005                                  ;;**********************************************************************
7006                                  ;*TEST 334       ROL DM1 TEST - N:C = 0101
7007                                  ;;**********************************************************************
7008  022640                          TST334:
7009  022640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7010  022642  012700  000334                  MOV     #334,R0         ;;LOAD R0 WITH TEST NUMBER
7011  022646  013701  022672                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7012  022652  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7013  022656  012704  125253                  MOV     #125253,R4      ;RESULT S / B = 125253
7014  022662  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
7015  022666  000257                          CCC                     ;CLEAR CODES
7016  022670  000265                          265                     ;N:C = 0101
7017
7018  022672  006112                  2$:     ROL     (R2)            ;TEST THE ROL
7019
7020  022674  100003                          BPL     3$              ;N:C = 1010 ?
7021  022676  001402                          BEQ     3$
7022  022700  102001                          BVC     3$
7023  022702  103001                          BCC     4$
7024
7025  022704  104001                  3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7026  022706  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7027  022710  001402                          BEQ     TST335          ;;BR IF YES
7028
```

```
7029  022712  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7030  022714  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7031
7032                          ;;**********************************************************************
7033                          ;*TEST 335       ROL DM1 TEST - N:C = 0010
7034                          ;;**********************************************************************
7035  022716                  TST335:
7036  022716  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7037  022720  012700  000335          MOV     #335,R0         ;;LOAD R0 WITH TEST NUMBER
7038  022724  013701  022744          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7039  022730  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7040  022734  005004                  CLR     R4              ;RESULT S / B = 000000
7041  022736  005012                  CLR     (R2)            ;[DEST] = 000000
7042  022740  000257                  CCC                     ;CLEAR CODES
7043  022742  000262                  SEV                     ;N:C = 0010
7044
7045  022744  006112          2$:     ROL     (R2)            ;TEST THE ROL
7046
7047  022746  100403                  BMI     3$              ;N:C = 0100 ?
7048  022750  001002                  BNE     3$
7049  022752  102401                  BVS     3$
7050  022754  103001                  BCC     4$
7051
7052  022756  104001          3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7053  022760  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
7054  022762  001402                  BEQ     TST336          ;;BR IF YES
7055
7056  022764  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7057  022766  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7058
7059                          ;;**********************************************************************
7060                          ;*TEST 336       ADC DM1 TEST - N:C = 0101
7061                          ;;**********************************************************************
7062  022770                  TST336:
7063  022770  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7064  022772  012700  000336          MOV     #336,R0         ;;LOAD R0 WITH TEST NUMBER
7065  022776  013701  023022          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7066  023002  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7067  023006  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
7068  023012  012712  077777          MOV     #77777,(R2)     ;[DEST] = 77777
7069  023016  000257                  CCC                     ;CLEAR CODES
7070  023020  000265                  265                     ;N:C = 0101
7071
7072  023022  005512          2$:     ADC     (R2)            ;TEST THE ADC
7073
7074  023024  100003                  BPL     3$              ;N:C = 1010 ?
7075  023026  001402                  BEQ     3$
7076  023030  102001                  BVC     3$
7077  023032  103001                  BCC     4$
7078
7079  023034  104001          3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
7080  023036  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
7081  023040  001402                  BEQ     TST337          ;;BR IF YES
7082
7083  023042  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7084  023044  104001          5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 337      ADC DM1 TEST - N:C = 1011
                                ;;*****************************************************************
023046                          TST337:
023046  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023050  012700  000337                  MOV     #337,R0         ;;LOAD R0 WITH TEST NUMBER
023054  013701  023076                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023060  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023064  005004                          CLR     R4              ;RESULT S  /  B = 000000
023066  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
023072  000257                          CCC                     ;CLEAR CODES
023074  000273                          273                     ;N:C = 1011

023076  005512                  2$:     ADC     (R2)            ;TEST THE ADC

023100  100403                          BMI     3$              ;N:C = 0101 ?
023102  001002                          BNE     3$
023104  102401                          BVS     3$
023106  103401                          BCS     4$

023110  104001                  3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
023112  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023114  001402                          BEQ     TST340          ;;BR IF YES

023116  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023120  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 340      ADC DM1 TEST - N:C = 1010
                                ;;*****************************************************************
023122                          TST340:
023122  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023124  012700  000340                  MOV     #340,R0         ;;LOAD R0 WITH TEST NUMBER
023130  013701  023154                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023134  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023140  012704  177777                  MOV     #-1,R4          ;RESULT S  /  B = 177777
023144  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
023150  000257                          CCC                     ;CLEAR CODES
023152  000272                          272                     ;N:C = 1010

023154  005512                  2$:     ADC     (R2)            ;TEST THE ADC

023156  100003                          BPL     3$              ;N:C = 1000 ?
023160  001402                          BEQ     3$
023162  102401                          BVS     3$
023164  103001                          BCC     4$

023166  104001                  3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
023170  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023172  001402                          BEQ     TST341          ;;BR IF YES

023174  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023176  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
```

```
7141                                    ;*TEST 341      SBC DM1 TEST - N:C = 1011
7142                                    ;;*****************************************************************
7143  023200                            TST341:
7144  023200  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7145  023202  012700  000341                   MOV     #341,R0         ;;LOAD R0 WITH TEST NUMBER
7146  023206  013701  023230                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7147  023212  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7148  023216  005004                           CLR     R4              ;RESULT S / B = 000000
7149  023220  012712  000001                   MOV     #1,(R2)         ;[DEST] = +1
7150  023224  000257                           CCC                     ;CLEAR CODES
7151  023226  000273                           273                     ;N:C = 1011
7152
7153  023230  005612                  2$:      SBC     (R2)            ;TEST THE SBC
7154
7155  023232  100403                           BMI     3$              ;N:C = 0100 ?
7156  023234  001002                           BNE     3$
7157  023236  102401                           BVS     3$
7158  023240  103001                           BCC     4$
7159
7160  023242  104001                  3$:      ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7161  023244  020412                  4$:      CMP     R4,(R2)         ;RESULT OK ?
7162  023246  001402                           BEQ     TST342          ;;BR IF YES
7163
7164  023250  011203                           MOV     (R2),R3         ;GET THE WAS DATA
7165  023252  104001                  5$:      ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7166
7167                                    ;;*****************************************************************
7168                                    ;*TEST 342      SBC DM1 TEST - N:C = 0101
7169                                    ;;*****************************************************************
7170  023254                            TST342:
7171  023254  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7172  023256  012700  000342                   MOV     #342,R0         ;;LOAD R0 WITH TEST NUMBER
7173  023262  013701  023306                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7174  023266  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7175  023272  012704  077777                   MOV     #077777,R4      ;RESULT S / B = 077777
7176  023276  012712  100000                   MOV     #100000,(R2)    ;[DEST] = 100000
7177  023302  000257                           CCC                     ;CLEAR CODES
7178  023304  000265                           265                     ;N:C = 0101
7179
7180  023306  005612                  2$:      SBC     (R2)            ;TEST THE SBC
7181
7182  023310  100403                           BMI     3$              ;N:C = 0010 ?
7183  023312  001402                           BEQ     3$
7184  023314  102001                           BVC     3$
7185  023316  103001                           BCC     4$
7186
7187  023320  104001                  3$:      ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7188  023322  020412                  4$:      CMP     R4,(R2)         ;RESULT OK ?
7189  023324  001402                           BEQ     TST343          ;;BR IF YES
7190
7191  023326  011203                           MOV     (R2),R3         ;GET THE WAS DATA
7192  023330  104001                  5$:      ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7193
7194                                    ;;*****************************************************************
7195                                    ;*TEST 343      SBC DM1 TEST - N:C = 1110
7196                                    ;;*****************************************************************
```

```
7197  023332
7198  023332  000004                  TST343:
7199  023334  012700  000343                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7200  023340  013701  023364                  MOV     #343,R0         ;;LOAD R0 WITH TEST NUMBER
7201  023344  012702  063312                  MOV     2#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7202  023350  012704  000001                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7203  023354  012712  000001                  MOV     #1,R4           ;RESULT S / B = 1
7204  023360  000257                          MOV     #1,(R2)         ;[DEST] = 1
7205  023362  000276                          CCC                     ;CLEAR CODES
7206                                           276                     ;N:C = 1110
7207  023364  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7208
7209  023366  100403                          BMI     3$              ;N:C = 0000 ?
7210  023370  001402                          BEQ     3$
7211  023372  102401                          BVS     3$
7212  023374  103001                          BCC     4$
7213
7214  023376  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7215  023400  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7216  023402  001402                          BEQ     TST344          ;;BR IF YES
7217
7218  023404  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7219  023406  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7220
7221                                   ;;**********************************************************************
7222                                   ;*TEST 344      SBC DM1 TEST - N:C = 0111
7223                                   ;;**********************************************************************
7224  023410                          TST344:
7225  023410  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7226  023412  012700  000344                  MOV     #344,R0         ;;LOAD R0 WITH TEST NUMBER
7227  023416  013701  023440                  MOV     2#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7228  023422  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7229  023426  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
7230  023432  005012                          CLR     (R2)            ;[DEST] = 000000
7231  023434  000257                          CCC                     ;CLEAR CODES
7232  023436  000267                          267                     ;N:C = 0111
7233
7234  023440  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7235
7236  023442  100003                          BPL     3$              ;N:C = 1001 ?
7237  023444  001402                          BEQ     3$
7238  023446  102401                          BVS     3$
7239  023450  103401                          BCS     4$
7240
7241  023452  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7242  023454  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7243  023456  001402                          BEQ     TST345          ;;BR IF YES
7244
7245  023460  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7246  023462  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7247
7248                                   ;;**********************************************************************
7249                                   ;*TEST 345      NEGB - MODE 0 TEST - N:C = 0110
7250                                   ;;**********************************************************************
7251  023464                          TST345:
7252  023464  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
7253  023466  012700  000345          MOV     #345,R0         ;;LOAD R0 WITH TEST NUMBER
7254  023472  013701  023512          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7255  023476  012704  177776          MOV     #177776,R4      ;RESULT S / B = 376 (LO BYTE)
7256  023502  012703  177402          MOV     #177402,R3      ;[DEST] = 177402
7257  023506  000257                  CCC                     ;CLEAR FLAGS
7258  023510  000266                  266                     ;N:C = 0110
7259
7260  023512  105403          2$:     NEGB    R3              ;TEST THE NEGB
7261
7262  023514  100003                  BPL     3$              ;N:C = 1001
7263  023516  001402                  BEQ     3$
7264  023520  102401                  BVS     3$
7265  023522  103401                  BCS     4$
7266
7267  023524  104002          3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7268
7269  023526  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
7270  023530  001401                  BEQ     TST346          ;;BR IF YES
7271
7272  023532  104002          5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7273
7274                          ;;**************************************************************
7275                          ;*TEST 346       NEGB - MODE 0 TEST - N:C = 0011
7276                          ;;**************************************************************
7277  023534                  TST346:
7278  023534  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7279  023536  012700  000346          MOV     #346,R0         ;;LOAD R0 WITH TEST NUMBER
7280  023542  013701  023562          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7281  023546  012704  177400          MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7282  023552  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
7283  023556  000257                  CCC                     ;CLEAR FLAGS
7284  023560  000263                  263                     ;N:C = 0011
7285
7286  023562  105403          2$:     NEGB    R3              ;TEST THE NEGB
7287
7288  023564  100403                  BMI     3$              ;N:C = 0100
7289  023566  001002                  BNE     3$
7290  023570  102401                  BVS     3$
7291  023572  103001                  BCC     4$
7292
7293  023574  104002          3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7294
7295  023576  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
7296  023600  001401                  BEQ     TST347          ;;BR IF YES
7297
7298  023602  104002          5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7299
7300                          ;;**************************************************************
7301                          ;*TEST 347       NEGB - MODE 0 TEST - N:C = 1101
7302                          ;;**************************************************************
7303  023604                  TST347:
7304  023604  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7305  023606  012700  000347          MOV     #347,R0         ;;LOAD R0 WITH TEST NUMBER
7306  023612  013701  023632          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7307  023616  012704  177600          MOV     #177600,R4      ;RESULT S / B = 200 (LO BYTE)
7308  023622  012703  177600          MOV     #177600,R3      ;[DEST] = 177600
```

```
7309  023626  000257                          CCC                     ;CLEAR FLAGS
7310  023630  000275                          275                     ;N:C = 1101
7311
7312  023632  105403                  2$:     NEGB    R3              ;TEST THE NEGB
7313
7314  023634  100003                          BPL     3$              ;N:C = 1011
7315  023636  001402                          BEQ     3$
7316  023640  102001                          BVC     3$
7317  023642  103401                          BCS     4$
7318
7319  023644  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7320
7321  023646  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7322  023650  001401                          BEQ     TST350          ;;BR IF YES
7323
7324  023652  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7325
7326                                  ;;**************************************************************
7327                                  ;*TEST 350       CLRB - MODE 0 TEST - N:C = 1011
7328                                  ;;**************************************************************
7329  023654                          TST350:
7330  023654  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7331  023656  012700  000350                  MOV     #350,R0         ;;LOAD R0 WITH TEST NUMBER
7332  023662  013701  023702                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7333  023666  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7334  023672  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
7335  023676  000257                          CCC                     ;CLEAR FLAGS
7336  023700  000273                          273                     ;N:C = 1011
7337
7338  023702  105003                  2$:     CLRB    R3              ;TEST THE CLRB
7339
7340  023704  100403                          BMI     3$              ;N:C = 0100 ?
7341  023706  001002                          BNE     3$
7342  023710  102401                          BVS     3$
7343  023712  103001                          BCC     4$
7344
7345  023714  104002                  3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
7346
7347  023716  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
7348  023720  001401                          BEQ     TST351          ;;BR IF YES
7349
7350  023722  104002                  5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
7351
7352                                  ;;**************************************************************
7353                                  ;*TEST 351       CLRB - MODE 0 TEST - N:C = 0100
7354                                  ;;**************************************************************
7355  023724                          TST351:
7356  023724  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7357  023726  012700  000351                  MOV     #351,R0         ;;LOAD R0 WITH TEST NUMBER
7358  023732  013701  023752                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7359  023736  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7360  023742  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
7361  023746  000257                          CCC                     ;CLEAR FLAGS
7362  023750  000264                          SEZ                     ;N:C = 0100
7363
7364  023752  105003                  2$:     CLRB    R3              ;TEST THE CLRB
```

```
7365
7366  023754  100403                        BMI     3$              ;N:C = 0100 ?
7367  023756  001002                        BNE     3$
7368  023760  102401                        BVS     3$
7369  023762  103001                        BCC     4$
7370
7371  023764  104002              3$:       ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
7372
7373  023766  020403              4$:       CMP     R4,R3           ;RESULT CORRECT ?
7374  023770  001401                        BEQ     TST352          ;;BR IF YES
7375
7376  023772  104002              5$:       ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
7377
7378                                        ;;*****************************************************************
7379                                        ;*TEST 352      CLRB TEST - DM2 - ODD ADDRESS
7380                                        ;;*****************************************************************
7381  023774                                TST352:
7382  023774  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7383  023776  012700  000352                MOV     #352,R0         ;;LOAD R0 WITH TEST NUMBER
7384  024002  013701  024032                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7385  024006  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7386  024012  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
7387  024016  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7388  024022  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7389  024024  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
7390  024030  000257                        CCC                     ;SCOPE SYNC
7391
7392  024032  105023              2$:       CLRB    (R3)+           ;TEST THE CLRB
7393
7394  024034  022703  063314                CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
7395  024040  001401                        BEQ     4$              ;BR IF YES
7396
7397  024042  104005              3$:       ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7398
7399  024044  020415              4$:       CMP     R4,(R5)         ;CORRECT RESULT ?
7400  024046  001402                        BEQ     TST353          ;;BR IF YES
7401
7402  024050  011503                        MOV     (R5),R3         ;GET THE WAS DATA
7403  024052  104001              5$:       ERROR   1               ;CLRB DELIVERED WRONG RESULT
7404
7405                                        ;;*****************************************************************
7406                                        ;*TEST 353      CLRB TEST - DM1 - ODD ADDRESS
7407                                        ;;*****************************************************************
7408  024054                                TST353:
7409  024054  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7410  024056  012700  000353                MOV     #353,R0         ;;LOAD R0 WITH TEST NUMBER
7411  024062  013701  024112                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7412  024066  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7413  024072  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
7414  024076  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7415  024102  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7416  024104  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
7417  024110  000257                        CCC                     ;SCOPE SYNC
7418
7419  024112  105013              2$:       CLRB    (R3)            ;TEST THE CLRB
7420
```

```
7421  024114  020415                      CMP     R4,(R5)         ;CORRECT RESULT ?
7422  024116  001402                      BEQ     TST354          ;;BR IF YES
7423
7424  024120  011503                      MOV     (R5),R3         ;GET THE WAS DATA
7425  024122  104001              3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7426
7427                              ;;**************************************************************
7428                              ;*TEST 354      CLRB TEST - DM2 - EVEN ADDRESS
7429                              ;;**************************************************************
7430  024124                      TST354:
7431  024124  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7432  024126  012700  000354              MOV     #354,R0         ;;LOAD R0 WITH TEST NUMBER
7433  024132  013701  024156              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7434  024136  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7435  024142  012704  177400              MOV     #177400,R4      ;RESULT S / B = 177400
7436  024146  010203                      MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7437  024150  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
7438  024154  000257                      CCC                     ;SCOPE SYNC
7439
7440  024156  105023              2$:     CLRB    (R3)+           ;TEST THE CLRB
7441
7442  024160  022703  063313              CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7443  024164  001401                      BEQ     4$              ;BR IF YES
7444
7445  024166  104005              3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7446
7447  024170  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7448  024172  001402                      BEQ     TST355          ;;BR IF YES
7449
7450  024174  011203                      MOV     (R2),R3         ;GET THE WAS DATA
7451  024176  104001              5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7452
7453                              ;;**************************************************************
7454                              ;*TEST 355      CLRB TEST - DM1 - EVEN ADDRESS
7455                              ;;**************************************************************
7456  024200                      TST355:
7457  024200  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7458  024202  012700  000355              MOV     #355,R0         ;;LOAD R0 WITH TEST NUMBER
7459  024206  013701  024232              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7460  024212  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7461  024216  012704  177400              MOV     #177400,R4      ;RESULT S / B = 177400
7462  024222  010203                      MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7463  024224  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
7464  024230  000257                      CCC                     ;SCOPE SYNC
7465
7466  024232  105013              2$:     CLRB    (R3)            ;TEST THE CLRB
7467
7468  024234  020412                      CMP     R4,(R2)         ;CORRECT RESULT ?
7469  024236  001402                      BEQ     TST356          ;;BR IF YES
7470
7471  024240  011203                      MOV     (R2),R3         ;GET THE WAS DATA
7472  024242  104001              3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7473
7474                              ;;**************************************************************
7475                              ;*TEST 356      NEGB TEST - DM2 - ODD ADDRESS
7476                              ;;**************************************************************
```

```
7477  024244                          TST356:
7478  024244  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7479  024246  012700  000356                  MOV     #356,R0         ;;LOAD R0 WITH TEST NUMBER
7480  024252  013701  024302                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7481  024256  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7482  024262  012704  000777                  MOV     #777,R4         ;RESULT S / B = 777
7483  024266  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7484  024272  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7485  024274  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
7486  024300  000257                          CCC                     ;SCOPE SYNC
7487
7488  024302  105423                  2$:     NEGB    (R3)+           ;TEST THE NEGB
7489
7490  024304  022703  063314                  CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
7491  024310  001401                          BEQ     4$              ;BR IF YES
7492
7493  024312  104005                  3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7494
7495  024314  020415                  4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
7496  024316  001402                          BEQ     TST357          ;;BR IF YES
7497
7498  024320  011503                          MOV     (R5),R3         ;GET THE WAS DATA
7499  024322  104001                  5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7500
7501                                  ;;**************************************************************
7502                                  ;*TEST 357      NEGB TEST - DM1 - ODD ADDRESS
7503                                  ;;**************************************************************
7504  024324                          TST357:
7505  024324  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7506  024326  012700  000357                  MOV     #357,R0         ;;LOAD R0 WITH TEST NUMBER
7507  024332  013701  024374                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7508                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT7
7509  024336  032737  000200  063234          BIT     #BIT7,@#BPTLOC  ;BREAKPOINT HALT SET ??
7510  024344  001401                          BEQ     .+4             ;BR IF NOT
7511  024346  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
7512  024350  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7513  024354  012704  000777                  MOV     #777,R4         ;RESULT S / B = 777
7514  024360  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7515  024364  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7516  024366  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
7517  024372  000257                          CCC                     ;SCOPE SYNC
7518
7519  024374  105413                  2$:     NEGB    (R3)            ;TEST THE NEGB
7520
7521  024376  020415                          CMP     R4,(R5)         ;CORRECT RESULT ?
7522  024400  001402                          BEQ     TST360          ;;BR IF YES
7523
7524  024402  011503                          MOV     (R5),R3         ;GET THE WAS DATA
7525  024404  104001                  3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7526
7527                                  ;;**************************************************************
7528                                  ;*TEST 360      NEGB TEST - DM2 - EVEN ADDRESS
7529                                  ;;**************************************************************
7530  024406                          TST360:
7531  024406  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7532  024410  012700  000360                  MOV     #360,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
7533  024414  013701  024440          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7534  024420  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7535  024424  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
7536  024430  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7537  024432  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7538  024436  000257                  CCC                     ;SCOPE SYNC
7539
7540  024440  105423          2$:     NEGB    (R3)+           ;TEST THE NEGB
7541
7542  024442  022703  063313          CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7543  024446  001401                  BEQ     4$              ;BR IF YES
7544
7545  024450  104005          3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7546
7547  024452  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7548  024454  001402                  BEQ     TST361          ;;BR IF YES
7549
7550  024456  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7551  024460  104001          5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7552
7553                          ;;**************************************************************
7554                          ;*TEST 361      NEGB TEST - DM1 - EVEN ADDRESS
7555                          ;;**************************************************************
7556  024462                  TST361:
7557  024462  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7558  024464  012700  000361          MOV     #361,R0         ;;LOAD R0 WITH TEST NUMBER
7559  024470  013701  024514          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7560  024474  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7561  024500  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
7562  024504  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7563  024506  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7564  024512  000257                  CCC                     ;SCOPE SYNC
7565
7566  024514  105413          2$:     NEGB    (R3)            ;TEST THE NEGB
7567
7568  024516  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
7569  024520  001402                  BEQ     TST362          ;;BR IF YES
7570
7571  024522  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7572  024524  104001          3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7573
7574                          ;;**************************************************************
7575                          ;*TEST 362      ADD TEST - SM0,DM0 - N:C = 1010
7576                          ;;**************************************************************
7577  024526                  TST362:
7578  024526  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7579  024530  012700  000362          MOV     #362,R0         ;;LOAD R0 WITH TEST NUMBER
7580  024534  013701  024556          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7581  024540  005004                  CLR     R4              ;RESULT S / B = 000000
7582  024542  012705  177777          MOV     #-1,R5          ;SRC OPR = 177777
7583  024546  012703  000001          MOV     #+1,R3          ;[DEST] = +1
7584  024552  000257                  CCC                     ;CLEAR FLAGS
7585  024554  000272                  272                     ;N:C = 1010
7586
7587  024556  060503          2$:     ADD     R5,R3           ;TEST THE ADD
7588
```

```
7589  024560  100403                          BMI     3$              ;N:C = 0101
7590  024562  001002                          BNE     3$
7591  024564  102401                          BVS     3$
7592  024566  103401                          BCS     4$
7593
7594  024570  104002                  3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7595
7596  024572  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7597  024574  001401                          BEQ     TST363          ;;BR IF YES
7598
7599  024576  104002                  5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7600
7601                                  ;;**********************************************************************
7602                                  ;*TEST 363       ADD TEST - SM0,DM0 - N:C = 0101
7603                                  ;;**********************************************************************
7604  024600                          TST363:
7605  024600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7606  024602  012700  000363                  MOV     #363,R0         ;;LOAD R0 WITH TEST NUMBER
7607  024606  013701  024632                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7608  024612  012704  100006                  MOV     #100006,R4      ;RESULT S / B = 100006
7609  024616  012705  077777                  MOV     #77777,R5       ;SRC OPR = 77777
7610  024622  012703  000007                  MOV     #7,R3           ;[DEST] = 7
7611  024626  000257                          CCC                     ;CLEAR FLAGS
7612  024630  000265                          265                     ;N:C = 0101
7613
7614  024632  060503                  2$:     ADD     R5,R3           ;TEST THE ADD
7615
7616  024634  100003                          BPL     3$              ;N:C = 1010
7617  024636  001402                          BEQ     3$
7618  024640  102001                          BVC     3$
7619  024642  103001                          BCC     4$
7620
7621  024644  104002                  3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7622
7623  024646  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7624  024650  001401                          BEQ     TST364          ;;BR IF YES
7625
7626  024652  104002                  5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7627
7628                                  ;;**********************************************************************
7629                                  ;*TEST 364       ADD SM1,DM0 TEST
7630                                  ;;**********************************************************************
7631  024654                          TST364:
7632  024654  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7633  024656  012700  000364                  MOV     #364,R0         ;;LOAD R0 WITH TEST NUMBER
7634  024662  013701  024702                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7635  024666  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7636  024672  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7637  024676  005003                          CLR     R3              ;[DEST] = 0
7638  024700  000257                          CCC                     ;SCOPE SYNC
7639
7640  024702  061503                  2$:     ADD     (R5),R3         ;TEST THE ADD - SM1,DM0
7641
7642  024704  020403                          CMP     R4,R3           ;RESULT = #DWTA?
7643  024706  001401                          BEQ     4$              ;BR IF YES
7644
```

```
7645  024710  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7646
7647  024712  022705  063276          4$:     CMP     #ATA,R5         ;DID ADD CHANGE REG.
7648  024716  001401                          BEQ     TST365          ;;BR IF NOT
7649
7650  024720  104005                  5$:     ERROR   5               ;REG GOT MODIFIED
7651
7652                                  ;;**************************************************************
7653                                  ;*TEST 365      ADD SM2,DMO TEST
7654                                  ;;**************************************************************
7655  024722                          TST365:
7656  024722  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7657  024724  012700  000365                  MOV     #365,R0         ;;LOAD R0 WITH TEST NUMBER
7658  024730  013701  024750                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7659  024734  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7660  024740  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7661  024744  005003                          CLR     R3              ;[DEST] = 0
7662  024746  000257                          CCC                     ;SCOPE SYNC
7663
7664  024750  062503                  2$:     ADD     (R5)+,R3        ;TEST THE ADD - SM2,DMO
7665
7666  024752  020403                          CMP     R4,R3           ;RESULT = #DWTA
7667  024754  001401                          BEQ     4$              ;BR IF YES
7668
7669  024756  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7670
7671  024760  022705  063300          4$:     CMP     #ATA+2,R5       ;DID ADD AUTO INCREMENT SOURCE REG?
7672  024764  001401                          BEQ     TST366          ;;BR IF YES
7673
7674  024766  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7675
7676                                  ;;**************************************************************
7677                                  ;*TEST 366      ADD SM3,DMO TEST
7678                                  ;;**************************************************************
7679  024770                          TST366:
7680  024770  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7681  024772  012700  000366                  MOV     #366,R0         ;;LOAD R0 WITH TEST NUMBER
7682  024776  013701  025022                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7683  025002  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7684  025006  012705  063306                  MOV     #ATA+10,R5      ;R5 POINTS TO SOURCE ADDR
7685  025012  010437  063312                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7686  025016  005003                          CLR     R3              ;[DEST] = 0
7687  025020  000257                          CCC                     ;SCOPE SYNC
7688
7689  025022  063503                  2$:     ADD     @(R5)+,R3       ;TEST THE ADD - SM3,DMO
7690
7691  025024  020437  063312                  CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7692  025030  001401                          BEQ     4$              ;BR IF YES
7693
7694  025032  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7695
7696  025034  022705  063310          4$:     CMP     #ATA+12,R5      ;DID ADD AUTO INCREMENT SOURCE REG?
7697  025040  001401                          BEQ     TST367          ;;BR IF YES
7698
7699  025042  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7700
```

```
                                ;;*****************************************************************
                                ;*TEST 367       ADD SM4,DMO TEST
                                ;;*****************************************************************
025044                          TST367:
025044  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025046  012700  000367                  MOV     #367,R0         ;;LOAD R0 WITH TEST NUMBER
025052  013701  025072                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025056  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025062  012705  063300                  MOV     #ATA+2,R5       ;SOURCE ADDR = ATA
025066  005003                          CLR     R3              ;[DEST] = 0
025070  000257                          CCC                     ;SCOPE SYNC

025072  064503                  2$:     ADD     -(R5),R3        ;TEST THE ADD - SM4,DMO

025074  020403                          CMP     R4,R3           ;RESULT = #DWTA?
025076  001401                          BEQ     4$              ;BR IF YES

025100  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

025102  022705  063276          4$:     CMP     #ATA,R5         ;DID SOURCE REG GET DECREMENTED?
025106  001401                          BEQ     TST370          ;;BR IF YES

025110  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG

                                ;;*****************************************************************
                                ;*TEST 370       ADD SM5,DMO TEST
                                ;;*****************************************************************
025112                          TST370:
025112  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025114  012700  000370                  MOV     #370,R0         ;;LOAD R0 WITH TEST NUMBER
025120  013701  025144                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025124  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025130  012705  063310                  MOV     #ATA+12,R5      ;R5 POINTS TO SOURCE ADDR
025134  010437  063312                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
025140  005003                          CLR     R3              ;[DEST] = 0
025142  000257                          CCC                     ;SCOPE SYNC

025144  065503                  2$:     ADD     @-(R5),R3       ;TEST THE ADD - SM5,DMO

025146  020437  063312                  CMP     R4,@#MBUF0      ;RESULT = #DWTA?
025152  001401                          BEQ     4$              ;BR IF YES

025154  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

025156  022705  063306          4$:     CMP     #ATA+10,R5      ;DID ADD DECREMENT SOURCE REG?
025162  001401                          BEQ     TST371          ;;BR IF YES

025164  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.

                                ;;*****************************************************************
                                ;*TEST 371       ADD SM6,DMO TEST
                                ;;*****************************************************************
025166                          TST371:
025166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025170  012700  000371                  MOV     #371,R0         ;;LOAD R0 WITH TEST NUMBER
025174  013701  025214                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7757  025200  012704  063312          MOV     #MBUFO,R4       ;RESULT S / B = MBUFO
7758  025204  012705  063276          MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7759  025210  005003                  CLR     R3              ;[DEST] = 0
7760  025212  000257                  CCC                     ;SCOPE SYNC
7761
7762  025214  066503  000010  2$:     ADD     10(R5),R3       ;TEST THE ADD - SM6,DMO
7763
7764  025220  020403                  CMP     R4,R3           ;RESULT =MBUFO?
7765  025222  001401                  BEQ     TST372          ;;BR IF YES
7766
7767  025224  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7768
7769                          ;;**************************************************************
7770                          ;*TEST 372      ADD SM7,DMO TEST
7771                          ;;**************************************************************
7772  025226                  TST372:
7773  025226  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7774  025230  012700  000372          MOV     #372,R0         ;;LOAD R0 WITH TEST NUMBER
7775  025234  013701  025260          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7776  025240  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7777  025244  012705  063276          MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7778  025250  010437  063312          MOV     R4,@#MBUFO      ;[SOURCE] = #DWTA
7779  025254  005003                  CLR     R3              ;[DEST] = 0
7780  025256  000257                  CCC                     ;SCOPE SYNC
7781
7782  025260  067503  000010  2$:     ADD     @10(R5),R3      ;TEST THE ADD - SM7,DMO
7783
7784  025264  020403                  CMP     R4,R3           ;RESULT = #DWTA?
7785  025266  001401                  BEQ     TST373          ;;BR IF YES
7786
7787  025270  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7788
7789                          ;;**************************************************************
7790                          ;*TEST 373      ADD SM1,DM1 TEST
7791                          ;;**************************************************************
7792  025272                  TST373:
7793  025272  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7794  025274  012700  000373          MOV     #373,R0         ;;LOAD R0 WITH TEST NUMBER
7795  025300  013701  025324          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7796  025304  012702  063312          MOV     #MBUFO,R2       ;DEST ADDR = MBUFO
7797  025310  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7798  025314  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7799  025320  005012                  CLR     (R2)            ;[DEST] = 0
7800  025322  000257                  CCC                     ;SCOPE SYNC
7801
7802  025324  061512          2$:     ADD     (R5),(R2)       ;TEST THE ADD - SM1,DM1
7803
7804  025326  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7805  025330  001402                  BEQ     TST374          ;;BR IF YES
7806
7807  025332  011203                  MOV     (R2),R3         ;GET WAS DATA
7808  025334  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7809
7810                          ;;**************************************************************
7811                          ;*TEST 374      ADD SM2,DM1 TEST
7812                          ;;**************************************************************
```

```
7813  025336                          TST374:
7814  025336  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7815  025340  012700  000374                  MOV     #374,R0         ;;LOAD R0 WITH TEST NUMBER
7816  025344  013701  025370                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7817  025350  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7818  025354  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7819  025360  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7820  025364  005012                          CLR     (R2)            ;[DEST] = 0
7821  025366  000257                          CCC                     ;SCOPE SYNC
7822
7823  025370  062512                  2$:     ADD     (R5)+,(R2)      ;TEST THE ADD - SM2,DM1
7824
7825  025372  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
7826  025374  001402                          BEQ     TST375          ;;BR IF YES
7827
7828  025376  011203                          MOV     (R2),R3         ;GET WAS DATA
7829  025400  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7830
7831                                  ;;**************************************************************
7832                                  ;*TEST 375       ADD SM1,DM2 TEST
7833                                  ;;**************************************************************
7834  025402                          TST375:
7835  025402  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7836  025404  012700  000375                  MOV     #375,R0         ;;LOAD R0 WITH TEST NUMBER
7837  025410  013701  025436                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7838  025414  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7839  025420  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7840  025424  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7841  025430  010203                          MOV     R2,R3           ;[R3] = DEST ADDR
7842  025432  005012                          CLR     (R2)            ;[DEST] = 0
7843  025434  000257                          CCC                     ;SCOPE SYNC
7844
7845  025436  061523                  2$:     ADD     (R5),(R3)+      ;TEST THE ADD - SM1,DM2
7846
7847  025440  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
7848  025442  001406                          BEQ     4$              ;BR IF YES
7849
7850  025444  010337  063316                  MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
7851  025450  011203                          MOV     (R2),R3         ;GET WAS DATA
7852  025452  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7853
7854  025454  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
7855  025460  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG
7856  025464  001401                          BEQ     TST376          ;;BR IF YES
7857
7858  025466  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7859
7860                                  ;;**************************************************************
7861                                  ;*TEST 376       ADD SM2,DM2 TEST
7862                                  ;;**************************************************************
7863  025470                          TST376:
7864  025470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7865  025472  012700  000376                  MOV     #376,R0         ;;LOAD R0 WITH TEST NUMBER
7866  025476  013701  025524                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7867  025502  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7868  025506  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
```

```
7869  025512  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7870  025516  010203                  MOV     R2,R3           ;[R3] = DEST ADDR
7871  025520  005012                  CLR     (R2)            ;[DEST] = 0
7872  025522  000257                  CCC                     ;SCOPE SYNC
7873
7874  025524  062523          2$:     ADD     (R5)+,(R3)+     ;TEST THE ADD - SM2,DM2
7875
7876  025526  020412                  CMP     R4,(R2)         ;RESULT = #DWTA
7877  025530  001406                  BEQ     4$              ;BR IF YES
7878
7879  025532  010337  063316          MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
7880  025536  011203                  MOV     (R2),R3         ;GET WAS DATA
7881  025540  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7882
7883  025542  013703  063316          MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
7884  025546  022703  063314  4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG?
7885  025552  001401                  BEQ     TST377          ;;BR IF YES
7886
7887  025554  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7888
7889                          ;;**********************************************************************
7890                          ;*TEST 377       ADD SM1,DM3 TEST
7891                          ;;**********************************************************************
7892  025556                  TST377:
7893  025556  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7894  025560  012700  000377          MOV     #377,R0         ;;LOAD R0 WITH TEST NUMBER
7895  025564  013701  025614          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7896  025570  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7897  025574  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7898  025600  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7899  025604  012703  063306          MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
7900  025610  005012                  CLR     (R2)            ;[DEST] = 0
7901  025612  000257                  CCC                     ;SCOPE SYNC
7902
7903  025614  061533          2$:     ADD     (R5),@(R3)+     ;TEST THE ADD - SM1,DM3
7904
7905  025616  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7906  025620  001406                  BEQ     4$              ;BR IF YES
7907
7908  025622  010337  063316          MOV     R3,@#MBUF1      ;SAVE R3
7909  025626  011203                  MOV     (R2),R3         ;GET WAS DATA
7910  025630  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7911
7912  025632  013703  063316          MOV     @#MBUF1,R3      ;RESTORE R3
7913  025636  022703  063310  4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
7914  025642  001401                  BEQ     TST400          ;;BR IF YES
7915
7916  025644  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7917
7918                          ;;**********************************************************************
7919                          ;*TEST 400       ADD SM2,DM3 TEST
7920                          ;;**********************************************************************
7921  025646                  TST400:
7922  025646  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7923  025650  012700  000400          MOV     #400,R0         ;;LOAD R0 WITH TEST NUMBER
7924  025654  013701  025704          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7925  025660  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7926  025664  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7927  025670  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7928  025674  012703  063306          MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
7929  025700  005012                  CLR     (R2)            ;[DEST] = 0
7930  025702  000257                  CCC                     ;SCOPE SYNC
7931
7932  025704  062533          2$:     ADD     (R5)+,@(R3)+    ;TEST THE ADD - SM2,DM3
7933
7934  025706  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7935  025710  001406                  BEQ     4$              ;BR IF YES
7936
7937  025712  010337  063316          MOV     R3,@#MBUF1      ;SAVE R3
7938  025716  011203                  MOV     (R2),R3         ;GET WAS DATA
7939  025720  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7940
7941  025722  013703  063316          MOV     @#MBUF1,R3      ;RESTORE R3
7942  025726  022703  063310  4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
7943  025732  001401                  BEQ     TST401          ;;BR IF YES
7944
7945  025734  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7946
7947                          ;;**********************************************************************
7948                          ;*TEST 401       ADD SM1,DM4 TEST
7949                          ;;**********************************************************************
7950  025736                  TST401:
7951  025736  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7952  025740  012700  000401          MOV     #401,R0         ;;LOAD R0 WITH TEST NUMBER
7953  025744  013701  025774          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7954  025750  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7955  025754  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7956  025760  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7957  025764  012703  063314          MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
7958  025770  005012                  CLR     (R2)            ;[DEST] = 0
7959  025772  000257                  CCC                     ;SCOPE SYNC
7960
7961  025774  061543          2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM1,DM4
7962
7963  025776  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7964  026000  001406                  BEQ     4$              ;BR IF YES
7965
7966  026002  010337  063316          MOV     R3,@#MBUF1      ;SAVE R3
7967  026006  011203                  MOV     (R2),R3         ;GET WAS DATA
7968  026010  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7969
7970  026012  013703  063316          MOV     @#MBUF1,R3      ;RESTORE R3
7971  026016  020302          4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
7972  026020  001401                  BEQ     TST402          ;;BR IF YES
7973
7974  026022  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
7975
7976                          ;;**********************************************************************
7977                          ;*TEST 402       ADD SM2,DM4 TEST
7978                          ;;**********************************************************************
7979  026024                  TST402:
7980  026024  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 7981  026026  012700  000402                MOV     #402,R0         ;;LOAD R0 WITH TEST NUMBER
 7982  026032  013701  026062                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7983  026036  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7984  026042  012704  063322                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 7985  026046  012705  063276                MOV     #ATA,R5         ;SOURCE ADDR = ATA
 7986  026052  012703  063314                MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
 7987  026056  005012                        CLR     (R2)            ;[DEST] = 0
 7988  026060  000257                        CCC                     ;SCOPE SYNC
 7989
 7990  026062  061543              2$:       ADD     (R5),-(R3)      ;TEST THE ADD - SM2,DM4
 7991
 7992  026064  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
 7993  026066  001406                        BEQ     4$              ;BR IF YES
 7994
 7995  026070  010337  063316                MOV     R3,@#MBUF1      ;SAVE R3
 7996  026074  011203                        MOV     (R2),R3         ;GET WAS DATA
 7997  026076  104001              3$:       ERROR   1               ;ADD DELIVERED WRONG RESULT
 7998
 7999  026100  013703  063316                MOV     @#MBUF1,R3      ;RESTORE R3
 8000  026104  020302              4$:       CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
 8001  026106  001401                        BEQ     TST403          ;;BR IF YES
 8002
 8003  026110  104005              5$:       ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
 8004
 8005                                        ;;*********************************************************************
 8006                                        ;*TEST 403      ADD SM1,DM5 TEST
 8007                                        ;;*********************************************************************
 8008  026112                                TST403:
 8009  026112  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8010  026114  012700  000403                      MOV     #403,R0         ;;LOAD R0 WITH TEST NUMBER
 8011  026120  013701  026150                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8012  026124  012702  063312                      MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8013  026130  012704  063322                      MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 8014  026134  012705  063276                      MOV     #ATA,R5         ;SOURCE ADDR = ATA
 8015  026140  012703  063310                      MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
 8016  026144  005012                              CLR     (R2)            ;[DEST] = 0
 8017  026146  000257                              CCC                     ;SCOPE SYNC
 8018
 8019  026150  061553              2$:       ADD     (R5),@-(R3)     ;TEST THE ADD - SM1,DM5
 8020
 8021  026152  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
 8022  026154  001406                        BEQ     4$              ;BR IF YES
 8023
 8024  026156  010337  063316                MOV     R3,@#MBUF1      ;SAVE R3
 8025  026162  011203                        MOV     (R2),R3         ;GET WAS DATA
 8026  026164  104001              3$:       ERROR   1               ;ADD DELIVERED WRONG RESULT
 8027
 8028  026166  013703  063316                MOV     @#MBUF1,R3      ;RESTORE R3
 8029  026172  022703  063306      4$:       CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
 8030  026176  001401                        BEQ     TST404          ;;BR IF YES
 8031
 8032  026200  104005              5$:       ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
 8033
 8034                                        ;;*********************************************************************
 8035                                        ;*TEST 404      ADD SM2,DM5 TEST
 8036                                        ;;*********************************************************************
```

```
8037  026202                          TST404:
8038  026202  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
8039  026204  012700  000404                  MOV     #404,R0                 ;;LOAD R0 WITH TEST NUMBER
8040  026210  013701  026240                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
8041  026214  012702  063312                  MOV     #MBUF0,R2               ;DEST ADDR = MBUF0
8042  026220  012704  063322                  MOV     #DWTA,R4                ;RESULT S / B = #DWTA
8043  026224  012705  063276                  MOV     #ATA,R5                 ;SOURCE ADDR = ATA
8044  026230  012703  063310                  MOV     #ATA+12,R3              ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
8045  026234  005012                          CLR     (R2)                    ;[DEST] = 0
8046  026236  000257                          CCC                             ;SCOPE SYNC
8047
8048  026240  062553                  2$:     ADD     (R5)+,@-(R3)            ;TEST THE ADD - SM2,DM5
8049
8050  026242  020412                          CMP     R4,(R2)                 ;RESULT = #DWTA?
8051  026244  001406                          BEQ     4$                      ;BR IF YES
8052
8053  026246  010337  063316                  MOV     R3,@#MBUF1              ;SAVE R3
8054  026252  011203                          MOV     (R2),R3                 ;GET WAS DATA
8055  026254  104001                  3$:     ERROR   1                       ;ADD DELIVERED WRONG RESULT
8056
8057  026256  013703  063316                  MOV     @#MBUF1,R3              ;RESTORE R3
8058  026262  022703  063306          4$:     CMP     #ATA+10,R3              ;DID ADD DECREMENT DEST REG?
8059  026266  001401                          BEQ     TST405                  ;;BR IF YES
8060
8061  026270  104005                  5$:     ERROR   5                       ;ADD FAILED TO UPDATE DEST REG
8062
8063                                  ;;**************************************************************
8064                                  ;*TEST 405       ADD SM1,DM6 TEST
8065                                  ;;**************************************************************
8066  026272                          TST405:
8067  026272  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
8068  026274  012700  000405                  MOV     #405,R0                 ;;LOAD R0 WITH TEST NUMBER
8069  026300  013701  026330                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
8070  026304  012702  063316                  MOV     #MBUF0+4,R2             ;DEST ADDR = MBUF0+4
8071  026310  012704  063322                  MOV     #DWTA,R4                ;RESULT S / B = #DWTA
8072  026314  012705  063276                  MOV     #ATA,R5                 ;SOURCE ADDR = ATA
8073  026320  012703  063312                  MOV     #MBUF0,R3               ;[R3] = BASE DEST ADDR
8074  026324  005012                          CLR     (R2)                    ;[DEST] = 0
8075  026326  000257                          CCC                             ;SCOPE SYNC
8076
8077  026330  061563  000004          2$:     ADD     (R5),4(R3)              ;TEST THE ADD - SM1,DM6
8078
8079  026334  020412                          CMP     R4,(R2)                 ;RESULT = #DWTA?
8080  026336  001402                          BEQ     TST406                  ;;BR IF YES
8081
8082  026340  011203                          MOV     (R2),R3                 ;GET WAS DATA
8083  026342  104001                  3$:     ERROR   1                       ;ADD DELIVERED WRONG RESULT
8084
8085                                  ;;**************************************************************
8086                                  ;*TEST 406       ADD SM2,DM6 TEST
8087                                  ;;**************************************************************
8088  026344                          TST406:
8089  026344  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
8090  026346  012700  000406                  MOV     #406,R0                 ;;LOAD R0 WITH TEST NUMBER
8091  026352  013701  026402                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
8092  026356  012702  063316                  MOV     #MBUF0+4,R2             ;DEST ADDR = MBUF0+4
```

```
8093  026362  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8094  026366  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
8095  026372  012703  063312          MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
8096  026376  005012                  CLR     (R2)            ;[DEST] = 0
8097  026400  000257                  CCC                     ;SCOPE SYNC
8098
8099  026402  062563  000004  2$:     ADD     (R5)+,4(R3)     ;TEST THE ADD - SM2,DM6
8100
8101  026406  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
8102  026410  001402                  BEQ     TST407          ;;BR IF YES
8103
8104  026412  011203                  MOV     (R2),R3         ;GET WAS DATA
8105  026414  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8106
8107                          ;;**************************************************************
8108                          ;*TEST 407       ADD SM1,DM7 TEST
8109                          ;;**************************************************************
8110  026416                  TST407:
8111  026416  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8112  026420  012700  000407          MOV     #407,R0         ;;LOAD R0 WITH TEST NUMBER
8113  026424  013701  026452          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8114  026430  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8115  026434  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8116  026440  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
8117  026444  010503                  MOV     R5,R3           ;BASE DEST ADDR = ATA
8118  026446  005012                  CLR     (R2)            ;[DEST] = 0
8119  026450  000257                  CCC                     ;SCOPE SYNC
8120
8121  026452  061573  000010  2$:     ADD     (R5),@10(R3)    ;TEST THE ADD - SM1,DM7
8122
8123  026456  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
8124  026460  001402                  BEQ     TST410          ;;BR IF YES
8125
8126  026462  011203                  MOV     (R2),R3         ;GET WAS DATA
8127  026464  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8128
8129                          ;;**************************************************************
8130                          ;*TEST 410       ADD SM2,DM7 TEST
8131                          ;;**************************************************************
8132  026466                  TST410:
8133  026466  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8134  026470  012700  000410          MOV     #410,R0         ;;LOAD R0 WITH TEST NUMBER
8135  026474  013701  026522          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8136  026500  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8137  026504  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8138  026510  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
8139  026514  010503                  MOV     R5,R3           ;BASE DEST ADDR = ATA
8140  026516  005012                  CLR     (R2)            ;[DEST] = 0
8141  026520  000257                  CCC                     ;SCOPE SYNC
8142
8143  026522  062573  000010  2$:     ADD     (R5)+,@10(R3)   ;TEST THE ADD - SM2,DM7
8144
8145  026526  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
8146  026530  001402                  BEQ     TST411          ;;BR IF YES
8147
8148  026532  011203                  MOV     (R2),R3         ;GET WAS DATA
```

```
8149  026534  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8150
8151                                  ;;*****************************************************************
8152                                  ;*TEST 411       "XOR RA,R8" TEST - A=B=000000 N:C=1010
8153                                  ;;*****************************************************************
8154  026536                          TST411:
8155  026536  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8156  026540  012700  000411                 MOV     #411,R0         ;;LOAD R0 WITH TEST NUMBER
8157  026544  013701  026560                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8158  026550  005004                         CLR     R4              ;RESULT AND MASK = 000000
8159  026552  005003                         CLR     R3              ;[DEST] = 000000
8160  026554  000257                         CCC                     ;SCOPE SYNC
8161  026556  000272                         272                     ;MAKE N:C=1010
8162
8163  026560  074403                  2$:     XOR     R4,R3           ;TEST THE XOR
8164
8165  026562  100403                         BMI     3$              ;N:C=0100 ??
8166  026564  001002                         BNE     3$
8167  026566  102401                         BVS     3$
8168  026570  103001                         BCC     4$
8169
8170  026572  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8171
8172  026574  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8173  026576  001401                         BEQ     TST412          ;;BR IF YES
8174
8175  026600  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8176
8177                                  ;;*****************************************************************
8178                                  ;*TEST 412       "XOR RA,RB" TEST - A=B=177777 N:C=0101
8179                                  ;;*****************************************************************
8180  026602                          TST412:
8181  026602  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8182  026604  012700  000412                 MOV     #412,R0         ;;LOAD R0 WITH TEST NUMBER
8183  026610  013701  026630                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8184  026614  005004                         CLR     R4              ;RESULT  = 000000
8185  026616  012705  177777                 MOV     #-1,R5          ;MASK = 177777
8186  026622  010503                         MOV     R5,R3           ;[DEST]=177777
8187  026624  000257                         CCC                     ;SCOPE SYNC
8188  026626  000265                         265                     ;MAKE N:C=0101
8189
8190  026630  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
8191
8192  026632  100403                         BMI     3$              ;N:C=0101 ??
8193  026634  001002                         BNE     3$
8194  026636  102401                         BVS     3$
8195  026640  103401                         BCS     4$
8196
8197  026642  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8198
8199  026644  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8200  026646  001401                         BEQ     TST413          ;;BR IF YES
8201
8202  026650  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8203
8204                                  ;;*****************************************************************
```

```
8205                                    ;*TEST 413      "XOR RA,RB" TEST - A=125252,B=052525 N:C=0110
8206                                    ;;*****************************************************************
8207  026652                            TST413:
8208  026652  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8209  026654  012700  000413                    MOV     #413,R0         ;;LOAD R0 WITH TEST NUMBER
8210  026660  013701  026704                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8211  026664  012704  177777                    MOV     #-1,R4          ;RESULT S/B = 177777
8212  026670  012705  125252                    MOV     #125252,R5      ;MASK=125252
8213  026674  012703  052525                    MOV     #052525,R3      ;[DEST] = 052525
8214  026700  000257                            CCC                     ;SCOPE SYNC
8215  026702  000266                            266                     ;MAKE N:C=0110
8216
8217  026704  074503                    2$:     XOR     R5,R3           ;TEST THE XOR
8218
8219  026706  100003                            BPL     3$              ;N:C=1000 ??
8220  026710  001402                            BEQ     3$
8221  026712  102401                            BVS     3$
8222  026714  103001                            BCC     4$
8223
8224  026716  104002                    3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8225
8226  026720  020403                    4$:     CMP     R4,R3           ;RESULT CORRECT?
8227  026722  001401                            BEQ     TST414          ;;BR IF YES
8228
8229  026724  104002                    5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8230
8231                                    ;;*****************************************************************
8232                                    ;*TEST 414      "XOR RA,RB" TEST - A=052525,B=125252 N:C=1001
8233                                    ;;*****************************************************************
8234  026726                            TST414:
8235  026726  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8236  026730  012700  000414                    MOV     #414,R0         ;;LOAD R0 WITH TEST NUMBER
8237  026734  013701  026760                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8238  026740  012704  177777                    MOV     #-1,R4          ;RESULT S/B = 177777
8239  026744  012705  052525                    MOV     #52525,R5       ;MASK=052525
8240  026750  012703  125252                    MOV     #125252,R3      ;[DEST] = 125252
8241  026754  000257                            CCC                     ;SCOPE SYNC
8242  026756  000271                            271                     ;MAKE N:C=1001
8243
8244  026760  074503                    2$:     XOR     R5,R3           ;TEST THE XOR
8245
8246  026762  100003                            BPL     3$              ;N:C=1001 ??
8247  026764  001402                            BEQ     3$
8248  026766  102401                            BVS     3$
8249  026770  103401                            BCS     4$
8250
8251  026772  104002                    3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8252
8253  026774  020403                    4$:     CMP     R4,R3           ;RESULT CORRECT?
8254  026776  001401                            BEQ     TST415          ;;BR IF YES
8255
8256  027000  104002                    5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8257
8258                                    ;;*****************************************************************
8259                                    ;*TEST 415      "XOR RA,(RB)" TEST - A=B=000000 N:C=1010
8260                                    ;;*****************************************************************
```

```
                                TST415:
8261  027002                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8262  027002  000004                    MOV     #415,R0         ;;LOAD R0 WITH TEST NUMBER
8263  027004  012700  000415            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8264  027010  013701  027032            CLR     R4              ;RESULT S / B = 000000
8265  027014  005004                    CLR     R5              ;MASK = 000000
8266  027016  005005                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8267  027020  012702  063312            CLR     (R2)            ;[DEST] = 000000
8268  027024  005012                    CCC                     ;SCOPE SYNC
8269  027026  000257                    272                     ;MAKE N:C=1010
8270  027030  000272
8271
8272  027032  074512            2$:     XOR     R5,(R2)         ;TEST THE XOR
8273
8274  027034  100403                    BMI     3$              ;N:C = 0100 ??
8275  027036  001002                    BNE     3$
8276  027040  102401                    BVS     3$
8277  027042  103001                    BCC     4$
8278
8279  027044  104001            3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8280
8281  027046  020412            4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8282  027050  001402                    BEQ     TST416          ;;BR IF YES
8283
8284  027052  011203                    MOV     (R2),R3         ;GET THE WAS DATA
8285  027054  104001            5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8286
8287                            ;;**********************************************************************
8288                            ;*TEST 416      "XOR RA,(RB)" TEST - A=B=177777 N:C=0101
8289                            ;;**********************************************************************
8290  027056                    TST416:
8291  027056  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8292  027060  012700  000416            MOV     #416,R0         ;;LOAD R0 WITH TEST NUMBER
8293  027064  013701  027112            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8294  027070  005004                    CLR     R4              ;RESULT S / B = 000000
8295  027072  012705  177777            MOV     #-1,R5          ;MASK = 177777
8296  027076  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8297  027102  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
8298  027106  000257                    CCC                     ;SCOPE SYNC
8299  027110  000265                    265                     ;MAKE N:C=0101
8300
8301  027112  074512            2$:     XOR     R5,(R2)         ;TEST THE XOR
8302
8303  027114  100403                    BMI     3$              ;N:C = 0101 ??
8304  027116  001002                    BNE     3$
8305  027120  102401                    BVS     3$
8306  027122  103401                    BCS     4$
8307
8308  027124  104001            3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8309
8310  027126  020412            4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8311  027130  001402                    BEQ     TST417          ;;BR IF YES
8312
8313  027132  011203                    MOV     (R2),R3         ;GET THE WAS DATA
8314  027134  104001            5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8315
8316                            ;;**********************************************************************
```

```
                                    ;*TEST 417      "XOR RA,(RB)" TEST - A=125252,B=052525 N:C=0110
                                    ;;*****************************************************************
027136                              TST417:
027136  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027140  012700  000417                      MOV     #417,R0         ;;LOAD R0 WITH TEST NUMBER
027144  013701  027206                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT8
027150  032737  000400  063234              BIT     #BIT8,@#BPTLOC  ;BREAKPOINT HALT SET ??
027156  001401                              BEQ     .+4             ;BR IF NOT
027160  000000                              HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
027162  012704  177777                      MOV     #-1,R4          ;RESULT S/B = 177777
027166  012705  125252                      MOV     #125252,R5      ;MASK = 125252
027172  012702  063312                      MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027176  012712  052525                      MOV     #052525,(R2)    ;[DEST] = 052525
027202  000257                              CCC                     ;SCOPE SYNC
027204  000266                              266                     ;MAKE N:C=0110
                                    
027206  074512                    2$:       XOR     R5,(R2)         ;TEST THE XOR
                                    
027210  100003                              BPL     3$              ;N:C = 1000 ??
027212  001402                              BEQ     3$
027214  102401                              BVS     3$
027216  103001                              BCC     4$
                                    
027220  104001                    3$:       ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
                                    
027222  020412                    4$:       CMP     R4,(R2)         ;RESULT CORRECT?
027224  001402                              BEQ     TST420          ;;BR IF YES
                                    
027226  011203                              MOV     (R2),R3         ;GET THE WAS DATA
027230  104001                    5$:       ERROR   1               ;XOR DELIVERED THE WRONG RESULT
                                    
                                    ;;*****************************************************************
                                    ;*TEST 420      "XOR RA,(RB)" TEST - A=052525,B=125252 N:C=1001
                                    ;;*****************************************************************
027232                              TST420:
027232  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027234  012700  000420                      MOV     #420,R0         ;;LOAD R0 WITH TEST NUMBER
027240  013701  027270                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027244  012704  177777                      MOV     #-1,R4          ;RESULT S/B = 177777
027250  012705  052525                      MOV     #52525,R5       ;MASK = 052525
027254  012702  063312                      MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027260  012712  125252                      MOV     #125252,(R2)    ;[DEST] = 125252
027264  000257                              CCC                     ;SCOPE SYNC
027266  000271                              271                     ;MAKE N:C=1001
                                    
027270  074512                    2$:       XOR     R5,(R2)         ;TEST THE XOR
                                    
027272  100003                              BPL     3$              ;N:C = 1001 ??
027274  001402                              BEQ     3$
027276  102401                              BVS     3$
027300  103401                              BCS     4$
                                    
027302  104001                    3$:       ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
                                    
027304  020412                    4$:       CMP     R4,(R2)         ;RESULT CORRECT?
```

```
8373  027306  001402                        BEQ     TST421          ;;BR IF YES
8374
8375  027310  011203                        MOV     (R2),R3         ;GET THE WAS DATA
8376  027312  104001                  5$:   ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8377
8378                                    ;;**************************************************************
8379                                    ;*TEST 421      SUB TEST SMD,DMD - (SRC) = (DEST) = +,+
8380                                    ;;**************************************************************
8381  027314                            TST421:
8382  027314  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8383  027316  012700  000421                MOV     #421,R0         ;;LOAD R0 WITH TEST NUMBER
8384  027322  013701  027342                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8385  027326  005004                        CLR     R4              ;RESULT S / B = 0
8386  027330  012703  052525                MOV     #052525,R3      ;[R3] = DEST OP = 52525
8387  027334  010305                        MOV     R3,R5           ;[R5] = SRC OP = 52525
8388  027336  000257                        CCC                     ;CLEAR FLAGS
8389  027340  000273                        273                     ;MAKE N:C = 1011
8390
8391  027342  160503                  2$:   SUB     R5,R3           ;TEST THE SUB
8392
8393  027344  100403                        BMI     3$
8394  027346  001002                        BNE     3$              ;DID N:C = 0100
8395  027350  102401                        BVS     3$
8396  027352  103001                        BCC     4$
8397
8398  027354  104002                  3$:   ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8399
8400  027356  020304                  4$:   CMP     R3,R4           ;WAS RESULT = 0?
8401  027360  001401                        BEQ     TST422          ;;BR IF YES
8402
8403  027362  104002                  5$:   ERROR   2               ;SUB DELIVERED WRONG RESULT
8404
8405                                    ;;**************************************************************
8406                                    ;*TEST 422      SUB TEST SMD,DMD - (SRC) = (DEST) = -,-
8407                                    ;;**************************************************************
8408  027364                            TST422:
8409  027364  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8410  027366  012700  000422                MOV     #422,R0         ;;LOAD R0 WITH TEST NUMBER
8411  027372  013701  027412                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8412  027376  005004                        CLR     R4              ;RESULT S / B = 0
8413  027400  012703  125252                MOV     #125252,R3      ;[R3] = DEST OP = 125252
8414  027404  010305                        MOV     R3,R5           ;[R5] = SOURCE OP = 125252
8415  027406  000257                        CCC                     ;CLEAR FLAGS
8416  027410  000273                        273                     ;MAKE N:C = 1011
8417
8418  027412  160503                  2$:   SUB     R5,R3           ;TEST THE SUB
8419
8420  027414  100403                        BMI     3$
8421  027416  001002                        BNE     3$              ;N:C = 0100?
8422  027420  102401                        BVS     3$
8423  027422  103001                        BCC     4$
8424
8425  027424  104002                  3$:   ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8426
8427  027426  020304                  4$:   CMP     R3,R4           ;RESULT = 0?
8428  027430  001401                        BEQ     TST423          ;;BR IF YES
```

```
8429
8430  027432  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8431
8432                                  ;;*****************************************************************
8433                                  ;*TEST 423      SUB TEST SMD,DMD - (SRC) = (DEST) = -,+
8434                                  ;;*****************************************************************
8435  027434                          TST423:
8436  027434  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8437  027436  012700  000423                  MOV     #423,R0         ;;LOAD R0 WITH TEST NUMBER
8438  027442  013701  027466                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8439  027446  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
8440  027452  012703  000001                  MOV     #1,R3           ;[R3] = DEST OP = 1
8441  027456  012705  177777                  MOV     #-1,R5          ;[R5] = SRC OP = -1
8442  027462  000257                          CCC                     ;CLEAR FLAGS
8443  027464  000276                          276                     ;MAKE N:C = 1110
8444
8445  027466  160503                  2$:     SUB     R5,R3           ;TEST THE SUB
8446
8447  027470  100403                          BMI     3$
8448  027472  001402                          BEQ     3$              ;N:C = 0001
8449  027474  102401                          BVS     3$
8450  027476  103401                          BCS     4$
8451
8452  027500  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8453
8454  027502  020304                  4$:     CMP     R3,R4           ;RESULT = +2?
8455  027504  001401                          BEQ     TST424          ;;BR IF YES
8456
8457  027506  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8458
8459                                  ;;*****************************************************************
8460                                  ;*TEST 424      SUB TEST SMD,DMD (SRC) = -(DEST) = +,-
8461                                  ;;*****************************************************************
8462  027510                          TST424:
8463  027510  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8464  027512  012700  000424                  MOV     #424,R0         ;;LOAD R0 WITH TEST NUMBER
8465  027516  013701  027542                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8466  027522  012704  177776                  MOV     #-2,R4          ;RESULT S / B = -2
8467  027526  012703  177777                  MOV     #-1,R3          ;[R3] = [DEST] = -1
8468  027532  012705  000001                  MOV     #1,R5           ;[R5] = [SOURCE] = +1
8469  027536  000257                          CCC                     ;CLEAR FLAGS
8470  027540  000267                          267                     ;MAKE N:C = 0111
8471
8472  027542  160503                  2$:     SUB     R5,R3           ;TEST THE SUB
8473
8474  027544  100003                          BPL     3$
8475  027546  001402                          BEQ     3$              ;N:C = 1000
8476  027550  102401                          BVS     3$
8477  027552  103001                          BCC     4$
8478
8479  027554  104002                  3$:     ERROR   2               ;SUB DID NOT ALTER CODES PROPERLY
8480
8481  027556  020403                  4$:     CMP     R4,R3           ;RESULT = -2?
8482  027560  001401                          BEQ     TST425          ;;BR IF YES
8483
8484  027562  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 425       SUB TEST SMO,DMO - "V" BIT SETS
                                ;;*****************************************************************
027564                          TST425:
027564  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027566  012700  000425                MOV     #425,R0         ;;LOAD R0 WITH TEST NUMBER
027572  013701  027616                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027576  012704  077777                MOV     #77777,R4       ;RESULT = 77777
027602  012703  100000                MOV     #100000,R3      ;[R3] = DEST OP = 100000
027606  012705  000001                MOV     #1,R5           ;[R5] = SRC OP = 1
027612  000257                        CCC                     ;CLEAR FLAGS
027614  000274                        274                     ;MAKE N:C = 1100

027616  160503                2$:     SUB     R5,R3           ;TEST THE SUB

027620  100403                        BMI     3$
027622  001402                        BEQ     3$              ;N:C = 0011 ("V" BIT SHOULD SET)
027624  102001                        BVC     3$
027626  103001                        BCC     4$

027630  104002                3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY

027632  020304                4$:     CMP     R3,R4           ;RESULT = 77777?
027634  001401                        BEQ     TST426          ;;BR IF YES

027636  104002                5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 426       SUB TEST - SMO,DM1 - N:C = 0110
                                ;;*****************************************************************
027640                          TST426:
027640  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027642  012700  000426                MOV     #426,R0         ;;LOAD R0 WITH TEST NUMBER
027646  013701  027674                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027652  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027656  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
027662  012705  000001                MOV     #+1,R5          ;SRC OPR = +1
027666  005012                        CLR     (R2)            ;[DEST] = 000000
027670  000257                        CCC                     ;CLEAR FLAGS
027672  000266                        266                     ;N:C = 0110

027674  160512                2$:     SUB     R5,(R2)         ;TEST THE SUB

027676  100003                        BPL     3$              ;N:C = 1001
027700  001402                        BEQ     3$
027702  102401                        BVS     3$
027704  103401                        BCS     4$

027706  104001                3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY

027710  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
027712  001402                        BEQ     TST427          ;;BR IF YES

027714  011203                        MOV     (R2),R3         ;GET THE WAS DATA
027716  104001                5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 427      SUB TEST - SMO,DMI - N:C = 1010
                                ;;*****************************************************************
027720                          TST427:
027720  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027722  012700  000427                  MOV     #427,R0         ;;LOAD R0 WITH TEST NUMBER
027726  013701  027754                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027732  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027736  005004                          CLR     R4              ;RESULT S / B = 000000
027740  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
027744  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
027750  000257                          CCC                     ;CLEAR FLAGS
027752  000272                          272                     ;N:C = 1010
027754  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
027756  100403                          BMI     3$              ;N:C = 0100
027760  001002                          BNE     3$
027762  102401                          BVS     3$
027764  103001                          BCC     4$
027766  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
027770  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
027772  001402                          BEQ     TST430          ;;BR IF YES
027774  011203                          MOV     (R2),R3         ;GET THE WAS DATA
027776  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
                                ;;*****************************************************************
                                ;*TEST 430      SUB TEST - SMO,DMI - N:C = 0000
                                ;;*****************************************************************
030000                          TST430:
030000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030002  012700  000430                  MOV     #430,R0         ;;LOAD R0 WITH TEST NUMBER
030006  013701  030034                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030012  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030016  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
030022  012705  000001                  MOV     #+1,R5          ;SRC OPR = +1
030026  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
030032  000257                          CCC                     ;CLEAR FLAGS
030034  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
030036  100403                          BMI     3$              ;N:C = 0010
030040  001402                          BEQ     3$
030042  102001                          BVC     3$
030044  103001                          BCC     4$
030046  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
030050  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
030052  001402                          BEQ     TST431          ;;BR IF YES
030054  011203                          MOV     (R2),R3         ;GET THE WAS DATA
```

```
8597  030056  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8598
8599                                  ;;*****************************************************************
8600                                  ;*TEST 431       SUB TEST - SM1,DM0 - N:C = 0110
8601                                  ;;*****************************************************************
8602  030060                          TST431:
8603  030060  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8604  030062  012700  000431                  MOV     #431,R0         ;;LOAD R0 WITH TEST NUMBER
8605  030066  013701  030110                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8606  030072  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
8607  030076  012705  064034                  MOV     #DWTB+2,R5      ;SRC ADDR = DWTB+2
8608  030102  005003                          CLR     R3              ;[DEST] = 000000
8609  030104  000257                          CCC                     ;CLEAR FLAGS
8610  030106  000266                          266                     ;N:C = 0110
8611
8612  030110  161503                  2$:     SUB     (R5),R3         ;TEST THE SUB
8613
8614  030112  100003                          BPL     3$              ;N:C = 1001
8615  030114  001402                          BEQ     3$
8616  030116  102401                          BVS     3$
8617  030120  103401                          BCS     4$
8618
8619  030122  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8620
8621  030124  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8622  030126  001401                          BEQ     TST432          ;;BR IF YES
8623
8624  030130  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8625
8626                                  ;;*****************************************************************
8627                                  ;*TEST 432       SUB TEST - SM1,DM0 - N:C = 1010
8628                                  ;;*****************************************************************
8629  030132                          TST432:
8630  030132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8631  030134  012700  000432                  MOV     #432,R0         ;;LOAD R0 WITH TEST NUMBER
8632  030140  013701  030160                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8633  030144  005004                          CLR     R4              ;RESULT S / B = 000000
8634  030146  012705  063324                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
8635  030152  011503                          MOV     (R5),R3         ;[DEST] = 177777
8636  030154  000257                          CCC                     ;CLEAR FLAGS
8637  030156  000272                          272                     ;N:C = 1010
8638
8639  030160  161503                  2$:     SUB     (R5),R3 ;TEST THE SUB
8640
8641  030162  100403                          BMI     3$              ;N:C = 0100
8642  030164  001002                          BNE     3$
8643  030166  102401                          BVS     3$
8644  030170  103001                          BCC     4$
8645
8646  030172  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8647
8648  030174  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8649  030176  001401                          BEQ     TST433          ;;BR IF YES
8650
8651  030200  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8652
```

```
                                ;;*****************************************************************
                                ;*TEST 433      SUB TEST - SM1,DM0 - N:C = 0000
                                ;;*****************************************************************
030202                          TST433:
030202  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030204  012700  000433                  MOV     #433,R0         ;;LOAD R0 WITH TEST NUMBER
030210  013701  030236                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030214  012704  077777                  MOV     #77777,R4               ;RESULT S / B = 77777
030220  012705  063316                  MOV     #MBUF1,R5               ;SRC ADDR =MBUF1
030224  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
030230  012715  000001                  MOV     #+1,(R5)        ;SRC OPR = +1
030234  000257                          CCC                     ;CLEAR FLAGS

030236  161503          2$:     SUB     (R5),R3         ;TEST THE SUB

030240  100403                  BMI     3$              ;N:C = 0010
030242  001402                  BEQ     3$
030244  102001                  BVC     3$
030246  103001                  BCC     4$

030250  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY

030252  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
030254  001401                  BEQ     TST434          ;;BR IF YES

030256  104002          5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 434      SUB  SM1,DM1 TEST - N:C = 0110
                                ;;*****************************************************************
030260                          TST434:
030260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030262  012700  000434                  MOV     #434,R0         ;;LOAD R0 WITH TEST NUMBER
030266  013701  030320                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030272  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030276  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
030302  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
030306  012715  000001                  MOV     #+1,(R5)        ;[SOURCE] = 000001
030312  005012                          CLR     (R2)            ;[DEST] = 000000
030314  000257                          CCC                     ;CLEAR FLAGS
030316  000266                          266                     ;N:C = 0110

030320  161512          2$:     SUB     (R5),(R2)       ;TEST THE SUB

030322  100003                  BPL     3$              ;N:C = 1001 ?
030324  001402                  BEQ     3$
030326  102401                  BVS     3$
030330  103401                  BCS     4$

030332  104001          3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY

030334  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
030336  001402                  BEQ     TST435          ;;BR IF YES

030340  011203                  MOV     (R2),R3         ;GET THE WAS DATA
030342  104001          5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
```

```
8709
8710                                    ;;*****************************************************************
8711                                    ;*TEST 435      SUB  SM1,DM2 TEST - N:C = 0110
8712                                    ;;*****************************************************************
8713  030344                            TST435:
8714  030344  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8715  030346  012700  000435                   MOV     #435,R0         ;;LOAD R0 WITH TEST NUMBER
8716  030352  013701  030406                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8717  030356  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8718  030362  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
8719  030366  012705  063316                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
8720  030372  012715  000001                   MOV     #+1,(R5)        ;[SOURCE] = 000001
8721  030376  005012                           CLR     (R2)            ;[DEST] = 000000
8722  030400  010203                           MOV     R2,R3           ;R3 GETS DEST ADDR
8723  030402  000257                           CCC                     ;CLEAR FLAGS
8724  030404  000266                           266                     ;N:C = 0110
8725
8726  030406  161523                   2$:     SUB     (R5),(R3)+      ;TEST THE SUB
8727
8728  030410  100003                           BPL     3$              ;N:C = 1001 ?
8729  030412  001402                           BEQ     3$
8730  030414  102401                           BVS     3$
8731  030416  103401                           BCS     4$
8732
8733  030420  104005                   3$:     ERROR   5               ;SUB FAILED TO ALTER CODES PROPERLY
8734
8735  030422  020412                   4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8736  030424  001402                           BEQ     TST436          ;;BR IF YES
8737
8738  030426  011203                           MOV     (R2),R3         ;GET THE WAS DATA
8739  030430  104001                   5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8740
8741                                    ;;*****************************************************************
8742                                    ;*TEST 436      NEG DM2 TEST
8743                                    ;;*****************************************************************
8744  030432                            TST436:
8745  030432  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8746  030434  012700  000436                   MOV     #436,R0         ;;LOAD R0 WITH TEST NUMBER
8747  030440  013701  030464                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8748  030444  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8749  030450  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8750  030454  010205                           MOV     R2,R5           ;[R5] = DEST ADDR
8751  030456  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8752  030462  000257                           CCC                     ;SCOPE SYNC
8753
8754  030464  005425                   2$:     NEG     (R5)+           ;TEST THE NEG - MODE 2
8755
8756  030466  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8757  030470  001402                           BEQ     4$              ;BR IF YES
8758
8759  030472  011203                           MOV     (R2),R3         ;GET THE WAS DATA
8760  030474  104001                   3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8761
8762  030476  022705  063314           4$:     CMP     #MBUF0+2,R5     ;DID REG. GET AUTO INCREMENTED?
8763  030502  001401                           BEQ     TST437          ;;BR IF YES
8764
```

```
8765  030504  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8766
8767                                  ;;*******************************************************************
8768                                  ;*TEST 437       NEG DM3 TEST
8769                                  ;;*******************************************************************
8770  030506                          TST437:
8771  030506  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8772  030510  012700  000437                 MOV     #437,R0         ;;LOAD R0 WITH TEST NUMBER
8773  030514  013701  030542                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8774  030520  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8775  030524  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
8776  030530  012705  063306                 MOV     #ATA+10,R5      ;[ATA+10] = MBUF0
8777  030534  012712  052526                 MOV     #52526,(R2)     ;[DEST] = 52526
8778  030540  000257                         CCC                     ;SCOPE SYNC
8779
8780  030542  005435                  2$:     NEG     @(R5)+          ;TEST THE NEG - MODE 3
8781
8782  030544  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8783  030546  001402                          BEQ     4$              ;BR IF YES
8784
8785  030550  011203                          MOV     (R2),R3         ;GET WAS DATA
8786  030552  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8787
8788  030554  022705  063310          4$:     CMP     #ATA+12,R5      ;DID REG GET AUTO INCREMENTED?
8789  030560  001401                          BEQ     TST440          ;;BR IF YES
8790
8791  030562  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8792
8793                                  ;;*******************************************************************
8794                                  ;*TEST 440       NEG DM4 TEST
8795                                  ;;*******************************************************************
8796  030564                          TST440:
8797  030564  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8798  030566  012700  000440                 MOV     #440,R0         ;;LOAD R0 WITH TEST NUMBER
8799  030572  013701  030620                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8800  030576  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8801  030602  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
8802  030606  012705  063314                 MOV     #MBUF0+2,R5     ;[R5] = DEST ADDR + 2
8803  030612  012712  052526                 MOV     #52526,(R2)     ;[DEST] = 52526
8804  030616  000257                         CCC                     ;SCOPE SYNC
8805
8806  030620  005445                  2$:     NEG     -(R5)           ;TEST THE NEG - MODE 4
8807
8808  030622  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8809  030624  001402                          BEQ     4$              ;BR IF YES
8810
8811  030626  011203                          MOV     (R2),R3         ;GET WAS DATA
8812  030630  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8813
8814  030632  020502                  4$:     CMP     R5,R2           ;DID REG GET AUTO INCREMENTED?
8815  030634  001401                          BEQ     TST441          ;;BR IF YES
8816
8817  030636  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
8818
8819                                  ;;*******************************************************************
8820                                  ;*TEST 441       NEG DM5 TEST
```

```
                                ;;*****************************************************************
030640                          TST441:
030640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030642  012700  000441                  MOV     #441,R0         ;;LOAD R0 WITH TEST NUMBER
030646  013701  030674                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030652  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030656  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
030662  012705  063310                  MOV     #ATA+12,R5      ;[R5] = (ADR OF MBUF0) +2
030666  012712  052526                  MOV     #52526,(R2)     ;[DEST] = 52526
030672  000257                          CCC                     ;SCOPE SYNC
                                
030674  005455                  2$:     NEG     @-(R5)          ;TEST THE NEG - MODE 5
                                
030676  020412                          CMP     R4,(R2)         ;RESULT = 125252?
030700  001402                          BEQ     4$              ;BR IF YES
                                
030702  011203                          MOV     (R2),R3         ;GET WAS DATA
030704  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
                                
030706  022705  063306          4$:     CMP     #ATA+10,R5      ;DID NEG UPDATE REG
030712  001401                          BEQ     TST442          ;;BR IF YES
                                
030714  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
                                
                                ;;*****************************************************************
                                ;;*TEST 442      NEG DM6 TEST
                                ;;*****************************************************************
030716                          TST442:
030716  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030720  012700  000442                  MOV     #442,R0         ;;LOAD R0 WITH TEST NUMBER
030724  013701  030752                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030730  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030734  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
030740  012705  063310                  MOV     #MBUF0-2,R5     ;[R5] = BASE ADDR
030744  012712  052526                  MOV     #52526,(R2)     ;[DEST] = 52526
030750  000257                          CCC                     ;SCOPE SYNC
                                
030752  005465  000002          2$:     NEG     2(R5)           ;TEST THE NEG - MODE 6
                                
030756  020412                          CMP     R4,(R2)         ;RESULT = 125252?
030760  001402                          BEQ     TST443          ;;BR IF YES
                                
030762  011203                          MOV     (R2),R3         ;GET WAS DATA
030764  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
                                
                                ;;*****************************************************************
                                ;;*TEST 443      NEG DM7 TEST
                                ;;*****************************************************************
030766                          TST443:
030766  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030770  012700  000443                  MOV     #443,R0         ;;LOAD R0 WITH TEST NUMBER
030774  013701  031022                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
031000  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
031004  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
031010  012705  063276                  MOV     #ATA,R5         ;[R5] = BASE ADDR
031014  012712  052526                  MOV     #52526,(R2)     ;[DEST] = 52526
```

```
8877  031020  000257                          CCC                     ;SCOPE SYNC
8878
8879  031022  005475  000010        2$:       NEG     @10(R5)         ;TEST THE NEG - MODE 7
8880
8881  031026  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8882  031030  001402                          BEQ     TST444          ;;BR IF YES
8883
8884  031032  011203                          MOV     (R2),R3         ;GET WAS DATA
8885  031034  104001        3$:       ERROR   1                       ;NEG DELIVERED WRONG RESULT
8886
8887                                  ;;**********************************************************************
8888                                  ;*TEST 444       MOV SM1,DM1 TEST - N:C = 0100
8889                                  ;;**********************************************************************
8890  031036                          TST444:
8891  031036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8892  031040  012700  000444                  MOV     #444,R0         ;;LOAD R0 WITH TEST NUMBER
8893  031044  013701  031074                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8894  031050  005004                          CLR     R4              ;RESULT S / B = 177777
8895  031052  005104                          COM     R4
8896  031054  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8897  031060  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8898  031064  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8899  031066  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8900  031070  000257                          CCC                     ;CLEAR FLAGS
8901  031072  000264                          264                     ;N:C = 0100
8902
8903  031074  011513        2$:       MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8904
8905  031076  100003                          BPL     3$              ;N:C = 1000 ?
8906  031100  001402                          BEQ     3$
8907  031102  102401                          BVS     3$
8908  031104  103001                          BCC     4$
8909
8910  031106  104001        3$:       ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8911
8912  031110  020412        4$:       CMP     R4,(R2)         ;RESULT CORRECT ??
8913  031112  001403                  BEQ     TST445          ;;BR IF YES
8914
8915  031114  005003                  CLR     R3              ;GET THE WAS DATA
8916  031116  051203                  BIS     (R2),R3
8917  031120  104001        5$:       ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8918
8919                                  ;;**********************************************************************
8920                                  ;*TEST 445       MOV SM2,DM1 TEST - N:C = 0100
8921                                  ;;**********************************************************************
8922  031122                          TST445:
8923  031122  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8924  031124  012700  000445          MOV     #445,R0         ;;LOAD R0 WITH TEST NUMBER
8925  031130  013701  031160          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8926  031134  005004                  CLR     R4              ;RESULT S / B = 177777
8927  031136  005104                  COM     R4
8928  031140  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8929  031144  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8930  031150  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8931  031152  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8932  031154  000257                  CCC                     ;CLEAR FLAGS
```

```
8933  031156  000264                          264                     ;N:C = 0100
8934
8935  031160  012513                  2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
8936
8937  031162  100003                          BPL     3$              ;N:C = 1000 ?
8938  031164  001402                          BEQ     3$
8939  031166  102401                          BVS     3$
8940  031170  103001                          BCC     4$
8941
8942  031172  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8943
8944  031174  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8945  031176  001403                          BEQ     TST446          ;;BR IF YES
8946
8947  031200  005003                          CLR     R3              ;GET THE WAS DATA
8948  031202  051203                          BIS     (R2),R3
8949  031204  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8950
8951                                  ;;**********************************************************************
8952                                  ;*TEST 446       MOV SM1,DM1 TEST - N:C = 1011
8953                                  ;;**********************************************************************
8954  031206                          TST446:
8955  031206  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8956  031210  012700  000446                  MOV     #446,R0         ;;LOAD R0 WITH TEST NUMBER
8957  031214  013701  031244                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8958  031220  005004                          CLR     R4              ;RESULT S / B = 000000
8959  031222  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8960  031226  012705  063322                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
8961  031232  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8962  031234  005012                          CLR     (R2)            ;MAKE [DEST] = 177777
8963  031236  005112                          COM     (R2)
8964  031240  000257                          CCC                     ;CLEAR FLAGS
8965  031242  000273                          273                     ;N:C = 1011
8966
8967  031244  011513                  2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8968
8969  031246  100403                          BMI     3$              ;N:C = 0101 ?
8970  031250  001002                          BNE     3$
8971  031252  102401                          BVS     3$
8972  031254  103401                          BCS     4$
8973
8974  031256  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8975
8976  031260  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8977  031262  001403                          BEQ     TST447          ;;BR IF YES
8978
8979  031264  005003                          CLR     R3              ;GET THE WAS DATA
8980  031266  051203                          BIS     (R2),R3
8981  031270  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8982
8983                                  ;;**********************************************************************
8984                                  ;*TEST 447       MOV SM2,DM1 TEST - N:C = 1011
8985                                  ;;**********************************************************************
8986  031272                          TST447:
8987  031272  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8988  031274  012700  000447                  MOV     #447,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
8989  031300  013701  031330          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8990  031304  005004                  CLR     R4              ;RESULT S / B = 000000
8991  031306  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8992  031312  012705  063322          MOV     #DATA,R5        ;SOURCE ADDR = DATA
8993  031316  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8994  031320  005012                  CLR     (R2)            ;MAKE [DEST] = 177777
8995  031322  005112                  COM     (R2)
8996  031324  000257                  CCC                     ;CLEAR FLAGS
8997  031326  000273                  273                     ;N:C = 1011
8998
8999  031330  012513          2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
9000
9001  031332  100403                  BMI     3$              ;N:C = 0101 ?
9002  031334  001002                  BNE     3$
9003  031336  102401                  BVS     3$
9004  031340  103401                  BCS     4$
9005
9006  031342  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9007
9008  031344  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9009  031346  001403                  BEQ     TST450          ;;BR IF YES
9010
9011  031350  005003                  CLR     R3              ;GET THE WAS DATA
9012  031352  051203                  BIS     (R2),R3
9013  031354  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9014
9015                          ;;*****************************************************************
9016                          ;*TEST 450       MOV SM1,DM2 TEST - N:C = 0100
9017                          ;;*****************************************************************
9018  031356                  TST450:
9019  031356  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9020  031360  012700  000450          MOV     #450,R0         ;;LOAD R0 WITH TEST NUMBER
9021  031364  013701  031414          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9022  031370  005004                  CLR     R4              ;RESULT S / B = 177777
9023  031372  005104                  COM     R4
9024  031374  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9025  031400  012705  063324          MOV     #DATA+2,R5      ;SOURCE ADDR = DATA
9026  031404  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
9027  031406  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
9028  031410  000257                  CCC                     ;CLEAR FLAGS
9029  031412  000264                  264                     ;N:C = 0100
9030
9031  031414  011523          2$:     MOV     (R5),(R3)+      ;TEST THE MOV - SM1,DM2
9032
9033  031416  100003                  BPL     3$              ;N:C =  1000 ?
9034  031420  001402                  BEQ     3$
9035  031422  102401                  BVS     3$
9036  031424  103001                  BCC     4$
9037
9038  031426  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9039
9040  031430  022703  063314  4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
9041  031434  001401                  BEQ     6$              ;BR IF YES
9042
9043  031436  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9044
```

```
9045  031440  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9046  031442  001403                          BEQ     TST451          ;;BR IF YES
9047
9048  031444  005003                          CLR     R3              ;GET THE WAS DATA
9049  031446  051203                          BIS     (R2),R3
9050  031450  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9051
9052                                  ;;**************************************************************
9053                                  ;*TEST 451       MOV SM2,DM2 TEST - N:C = 0100
9054                                  ;;**************************************************************
9055  031452                          TST451:
9056  031452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9057  031454  012700  000451                  MOV     #451,R0         ;;LOAD R0 WITH TEST NUMBER
9058  031460  013701  031510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9059  031464  005004                          CLR     R4              ;RESULT S / B = 177777
9060  031466  005104                          COM     R4
9061  031470  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9062  031474  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
9063  031500  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
9064  031502  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
9065  031504  000257                          CCC                     ;CLEAR FLAGS
9066  031506  000264                          264                     ;N:C = 0100
9067
9068  031510  012523                  2$:     MOV     (R5)+,(R3)+     ;TEST THE MOV - SM2,DM2
9069
9070  031512  100003                          BPL     3$              ;N:C = 1000 ?
9071  031514  001402                          BEQ     3$
9072  031516  102401                          BVS     3$
9073  031520  103001                          BCC     4$
9074
9075  031522  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9076
9077  031524  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
9078  031530  001401                          BEQ     6$              ;BR IF YES
9079
9080  031532  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9081
9082  031534  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9083  031536  001403                          BEQ     TST452          ;;BR IF YES
9084
9085  031540  005003                          CLR     R3              ;GET THE WAS DATA
9086  031542  051203                          BIS     (R2),R3
9087  031544  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9088
9089                                  ;;**************************************************************
9090                                  ;*TEST 452       MOV SM1,DM3 TEST - N:C = 0100
9091                                  ;;**************************************************************
9092  031546                          TST452:
9093  031546  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9094  031550  012700  000452                  MOV     #452,R0         ;;LOAD R0 WITH TEST NUMBER
9095  031554  013701  031606                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9096  031560  005004                          CLR     R4              ;RESULT S / B = 177777
9097  031562  005104                          COM     R4
9098  031564  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9099  031570  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9100  031574  012703  063306                  MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
```

```
 9101  031600  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
 9102  031602  000257                        CCC                     ;CLEAR FLAGS
 9103  031604  000264                        264                     ;N:C = 0100
 9104
 9105  031606  011533                2$:     MOV     (R5),@(R3)+     ;TEST THE MOV - SM1,DM3
 9106
 9107  031610  100003                        BPL     3$              ;N:C = 1000 ?
 9108  031612  001402                        BEQ     3$
 9109  031614  102401                        BVS     3$
 9110  031616  103001                        BCC     4$
 9111
 9112  031620  104001                3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9113
 9114  031622  022703  063310        4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9115  031626  001401                        BEQ     6$              ;BR IF YES
 9116
 9117  031630  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9118
 9119  031632  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9120  031634  001403                        BEQ     TST453          ;;BR IF YES
 9121
 9122  031636  005003                        CLR     R3              ;GET THE WAS DATA
 9123  031640  051203                        BIS     (R2),R3
 9124  031642  104001                7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9125
 9126                                ;;**************************************************************
 9127                                ;*TEST 453      MOV SM2,DM3 TEST - N:C = 0100
 9128                                ;;**************************************************************
 9129  031644                        TST453:
 9130  031644  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9131  031646  012700  000453                MOV     #453,R0         ;;LOAD R0 WITH TEST NUMBER
 9132  031652  013701  031704                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9133  031656  005004                        CLR     R4              ;RESULT S / B = 177777
 9134  031660  005104                        COM     R4
 9135  031662  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9136  031666  012705  063324                MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9137  031672  012703  063306                MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9138  031676  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
 9139  031700  000257                        CCC                     ;CLEAR FLAGS
 9140  031702  000264                        264                     ;N:C = 0100
 9141
 9142  031704  012533                2$:     MOV     (R5)+,@(R3)+    ;TEST THE MOV - SM2,DM3
 9143
 9144  031706  100003                        BPL     3$              ;N:C = 1000 ?
 9145  031710  001402                        BEQ     3$
 9146  031712  102401                        BVS     3$
 9147  031714  103001                        BCC     4$
 9148
 9149  031716  104001                3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9150
 9151  031720  022703  063310        4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9152  031724  001401                        BEQ     6$              ;BR IF YES
 9153
 9154  031726  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9155
 9156  031730  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
```

```
9157  031732  001403                          BEQ     TST454          ;;BR IF YES
9158
9159  031734  005003                          CLR     R3              ;GET THE WAS DATA
9160  031736  051203                          BIS     (R2),R3
9161  031740  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9162
9163                                  ;;**********************************************************************
9164                                  ;*TEST 454       MOV SM1,DM4 TEST - N:C = 0100
9165                                  ;;**********************************************************************
9166  031742                          TST454:
9167  031742  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9168  031744  012700  000454                  MOV     #454,R0         ;;LOAD R0 WITH TEST NUMBER
9169  031750  013701  032002                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9170  031754  005004                          CLR     R4              ;RESULT S / B = 177777
9171  031756  005104                          COM     R4
9172  031760  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9173  031764  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9174  031770  012703  063314                  MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
9175  031774  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
9176  031776  000257                          CCC                     ;CLEAR FLAGS
9177  032000  000264                          264                     ;N:C = 0100
9178
9179  032002  011543                  2$:     MOV     (R5),-(R3)      ;TEST THE MOV - SM1,DM4
9180
9181  032004  100003                          BPL     3$              ;N:C = 1000 ?
9182  032006  001402                          BEQ     3$
9183  032010  102401                          BVS     3$
9184  032012  103001                          BCC     4$
9185
9186  032014  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9187
9188  032016  020203                  4$:     CMP     R2,R3           ;DID MOV DECREMENT DEST REG ?
9189  032020  001401                          BEQ     6$              ;BR IF YES
9190
9191  032022  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9192
9193  032024  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9194  032026  001403                          BEQ     TST455          ;;BR IF YES
9195
9196  032030  005003                          CLR     R3              ;GET THE WAS DATA
9197  032032  051203                          BIS     (R2),R3
9198  032034  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9199
9200                                  ;;**********************************************************************
9201                                  ;*TEST 455       MOV SM2,DM4 TEST - N:C = 0100
9202                                  ;;**********************************************************************
9203  032036                          TST455:
9204  032036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9205  032040  012700  000455                  MOV     #455,R0         ;;LOAD R0 WITH TEST NUMBER
9206  032044  013701  032076                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9207  032050  005004                          CLR     R4              ;RESULT S / B = 177777
9208  032052  005104                          COM     R4
9209  032054  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9210  032060  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9211  032064  012703  063314                  MOV     #MBUF0+2,R3             ;BASE DEST ADDR = MBUF0+2
9212  032070  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
```

```
9213  032072  000257                          CCC                     ;CLEAR FLAGS
9214  032074  000264                          264                     ;N:C = 0100
9215
9216  032076  012543                  2$:     MOV     (R5)+,-(R3)     ;TEST THE MOV - SM2,DM4
9217
9218  032100  100003                          BPL     3$              ;N:C = 1000 ?
9219  032102  001402                          BEQ     3$
9220  032104  102401                          BVS     3$
9221  032106  103001                          BCC     4$
9222
9223  032110  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9224
9225  032112  020203                  4$:     CMP     R2,R3           ;DID MOV INCREMENT DEST REG ?
9226  032114  001401                          BEQ     6$              ;BR IF YES
9227
9228  032116  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9229
9230  032120  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9231  032122  001403                          BEQ     TST456          ;;BR IF YES
9232
9233  032124  005003                          CLR     R3              ;GET THE WAS DATA
9234  032126  051203                          BIS     (R2),R3
9235  032130  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9236
9237                                  ;;**************************************************************
9238                                  ;*TEST 456       MOV SM1,DM5 TEST - N:C = 0100
9239                                  ;;**************************************************************
9240  032132                          TST456:
9241  032132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9242  032134  012700  000456                  MOV     #456,R0         ;;LOAD R0 WITH TEST NUMBER
9243  032140  013701  032204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9244                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT9
9245  032144  032737  001000  063234          BIT     #BIT9,@#BPTLOC  ;BREAKPOINT HALT SET ??
9246  032152  001401                          BEQ     .+4             ;BR IF NOT
9247  032154  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
9248  032156  005004                          CLR     R4              ;RESULT S / B = 177777
9249  032160  005104                          COM     R4
9250  032162  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9251  032166  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9252  032172  012703  063310                  MOV     #ATA+12,R3              ;BASE DEST ADDR = ATA+12
9253  032176  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
9254  032200  000257                          CCC                     ;CLEAR FLAGS
9255  032202  000264                          264                     ;N:C = 0100
9256
9257  032204  011553                  2$:     MOV     (R5),@-(R3)     ;TEST THE MOV - SM1,DM5
9258
9259  032206  100003                          BPL     3$              ;N:C = 0100 ?
9260  032210  001402                          BEQ     3$
9261  032212  102401                          BVS     3$
9262  032214  103001                          BCC     4$
9263
9264  032216  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9265
9266  032220  022703  063306          4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
9267  032224  001401                          BEQ     6$              ;BR IF YES
9268
```

```
9269  032226  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9270
9271  032230  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9272  032232  001403                          BEQ     TST457          ;;BR IF YES
9273
9274  032234  005003                          CLR     R3              ;GET THE WAS DATA
9275  032236  051203                          BIS     (R2),R3
9276  032240  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9277
9278                                  ;;**************************************************************
9279                                  ;*TEST 457       MOV SM2,DM5 TEST - N:C = 0100
9280                                  ;;**************************************************************
9281  032242                          TST457:
9282  032242  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9283  032244  012700  000457                  MOV     #457,R0         ;;LOAD R0 WITH TEST NUMBER
9284  032250  013701  032302                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9285  032254  005004                          CLR     R4              ;RESULT S / B = 177777
9286  032256  005104                          COM     R4
9287  032260  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9288  032264  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
9289  032270  012703  063310                  MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
9290  032274  005012                          CLR     (R2)            ;MAKE [DEST] = 00000
9291  032276  000257                          CCC                     ;CLEAR FLAGS
9292  032300  000264                          264                     ;N:C = 1000
9293
9294  032302  012553                  2$:     MOV     (R5)+,@-(R3)    ;TEST THE MOV - SM2,DM5
9295
9296  032304  100003                          BPL     3$              ;N:C = 1000 ?
9297  032306  001402                          BEQ     3$
9298  032310  102401                          BVS     3$
9299  032312  103001                          BCC     4$
9300
9301  032314  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9302
9303  032316  022703  063306          4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
9304  032322  001401                          BEQ     6$              ;BR IF YES
9305
9306  032324  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
9307
9308  032326  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9309  032330  001403                          BEQ     TST460          ;;BR IF YES
9310
9311  032332  005003                          CLR     R3              ;GET THE WAS DATA
9312  032334  051203                          BIS     (R2),R3
9313  032336  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9314
9315                                  ;;**************************************************************
9316                                  ;*TEST 460       MOV SM1,DM6 TEST - N:C = 0100
9317                                  ;;**************************************************************
9318  032340                          TST460:
9319  032340  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9320  032342  012700  000460                  MOV     #460,R0         ;;LOAD R0 WITH TEST NUMBER
9321  032346  013701  032400                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9322  032352  005004                          CLR     R4              ;RESULT S / B = 177777
9323  032354  005104                          COM     R4
9324  032356  012702  063320                  MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
```

```
 9325  032362  012705  063324              MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9326  032366  012703  063312              MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9327  032372  005012                      CLR     (R2)            ;MAKE [DEST] = 000000
 9328  032374  000257                      CCC                     ;CLEAR FLAGS
 9329  032376  000264                      264                     ;N:C = 0100
 9330
 9331  032400  011563  000006      2$:     MOV     (R5),6(R3)      ;TEST THE MOV - SM1,DM6
 9332
 9333  032404  100003                      BPL     3$              ;N:C = 1000 ?
 9334  032406  001402                      BEQ     3$
 9335  032410  102401                      BVS     3$
 9336  032412  103001                      BCC     4$
 9337
 9338  032414  104001      3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9339
 9340  032416  020412      4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9341  032420  001403                      BEQ     TST461          ;;BR IF YES
 9342
 9343  032422  005003                      CLR     R3              ;GET THE WAS DATA
 9344  032424  051203                      BIS     (R2),R3
 9345  032426  104001      5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9346
 9347                              ;;*****************************************************************
 9348                              ;*TEST 461      MOV SM2,DM6 TEST - N:C = 0100
 9349                              ;;*****************************************************************
 9350  032430                      TST461:
 9351  032430  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9352  032432  012700  000461              MOV     #461,R0         ;;LOAD R0 WITH TEST NUMBER
 9353  032436  013701  032470              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9354  032442  005004                      CLR     R4              ;RESULT S / B = 177777
 9355  032444  005104                      COM     R4
 9356  032446  012702  063320              MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9357  032452  012705  063324              MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9358  032456  012703  063312              MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9359  032462  005012                      CLR     (R2)            ;MAKE [DEST] = 000000
 9360  032464  000257                      CCC                     ;CLEAR FLAGS
 9361  032466  000264                      264                     ;N:C = 0100
 9362
 9363  032470  012563  000006      2$:     MOV     (R5)+,6(R3)     ;TEST THE MOV - SM2,DM6
 9364
 9365  032474  100003                      BPL     3$              ;N:C = 1000 ?
 9366  032476  001402                      BEQ     3$
 9367  032500  102401                      BVS     3$
 9368  032502  103001                      BCC     4$
 9369
 9370  032504  104001      3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9371
 9372  032506  020412      4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9373  032510  001403                      BEQ     TST462          ;;BR IF YES
 9374
 9375  032512  005003                      CLR     R3              ;GET THE WAS DATA
 9376  032514  051203                      BIS     (R2),R3
 9377  032516  104001      5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9378
 9379                              ;;*****************************************************************
 9380                              ;*TEST 462      MOV SM1,DM7 TEST - N:C = 0100
```

```
                                ;;*****************************************************************
032520                          ;T462:
032520  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032522  012700  000462                 MOV     #462,R0         ;;LOAD R0 WITH TEST NUMBER
032526  013701  032560                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032532  005004                          CLR     R4              ;RESULT S / B = 177777
032534  005104                          COM     R4
032536  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032542  012705  063324                 MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032546  012703  063276                 MOV     #ATA,R3         ;BASE DEST ADDR = ATA
032552  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
032554  000257                          CCC                     ;CLEAR FLAGS
032556  000264                          264                     ;N:C = 0100
                                
032560  011573  000010         2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM1,DM7
                                
032564  100003                          BPL     3$              ;N:C = 1000 ?
032566  001402                          BEQ     3$
032570  102401                          BVS     3$
032572  103001                          BCC     4$
                                
032574  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
                                
032576  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
032600  001403                          BEQ     T$T463          ;;BR IF YES
                                
032602  005003                          CLR     R3              ;GET THE WAS DATA
032604  051203                          BIS     (R2),R3
032606  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;;TEST 463      MOV SM2,DM7 TEST - N:C = 0100
                                ;;*****************************************************************
032610                          T$T463:
032610  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032612  012700  000463                 MOV     #463,R0         ;;LOAD R0 WITH TEST NUMBER
032616  013701  032650                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032622  005004                          CLR     R4              ;RESULT S / B = 177777
032624  005104                          COM     R4
032626  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032632  012705  063324                 MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032636  012703  063276                 MOV     #ATA,R3         ;BASE DEST ADDR = ATA
032642  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
032644  000257                          CCC                     ;CLEAR FLAGS
032646  000264                          264                     ;N:C = 0100
                                
032650  011573  000010         2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM2,DM7
                                
032654  100003                          BPL     3$              ;N:C = 1000 ?
032656  001402                          BEQ     3$
032660  102401                          BVS     3$
032662  103001                          BCC     4$
                                
032664  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
                                
032666  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
```

```
9437  032670  001403                        BEQ     TST464          ;;BR IF YES
9438
9439  032672  005003                        CLR     R3              ;GET THE WAS DATA
9440  032674  051203                        BIS     (R2),R3
9441  032676  104001                5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9442
9443                                  ;;*****************************************************************
9444                                  ;*TEST 464      MOV  SM0,DM1 TEST
9445                                  ;;*****************************************************************
9446  032700                          TST464:
9447  032700  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9448  032702  012700  000464                MOV     #464,R0         ;;LOAD R0 WITH TEST NUMBER
9449  032706  013701  032726                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9450  032712  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9451  032716  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9452  032720  010205                        MOV     R2,R5           ;R5 GETS DEST ADDR
9453  032722  005012                        CLR     (R2)            ;[DEST] = 000000
9454  032724  000257                        CCC                     ;SCOPE SYNC
9455
9456  032726  010015                2$:     MOV     R0,(R5)         ;TEST THE MOV
9457
9458  032730  020412                        CMP     R4,(R2)         ;RESULT CORRECT ?
9459  032732  001402                        BEQ     TST465          ;;BR IF YES
9460
9461  032734  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9462  032736  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9463
9464                                  ;;*****************************************************************
9465                                  ;*TEST 465      MOV  SM0,DM2 TEST
9466                                  ;;*****************************************************************
9467  032740                          TST465:
9468  032740  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9469  032742  012700  000465                MOV     #465,R0         ;;LOAD R0 WITH TEST NUMBER
9470  032746  013701  032766                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9471  032752  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9472  032756  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9473  032760  010205                        MOV     R2,R5           ;R5 GETS DEST ADDR
9474  032762  005012                        CLR     (R2)            ;[DEST] = 000000
9475  032764  000257                        CCC                     ;SCOPE SYNC
9476
9477  032766  010025                2$:     MOV     R0,(R5)+        ;TEST THE MOV
9478
9479  032770  020412                        CMP     R4,(R2)         ;RESULT CORRECT ?
9480  032772  001402                        BEQ     TST466          ;;BR IF YES
9481
9482  032774  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9483  032776  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9484
9485                                  ;;*****************************************************************
9486                                  ;*TEST 466      MOV  SM0,DM3 TEST
9487                                  ;;*****************************************************************
9488  033000                          TST466:
9489  033000  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9490  033002  012700  000466                MOV     #466,R0         ;;LOAD R0 WITH TEST NUMBER
9491  033006  013701  033030                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9492  033012  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
9493  033016  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9494  033020  012705  063306                MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
9495  033024  005012                        CLR     (R2)            ;[DEST] = 000000
9496  033026  000257                        CCC                     ;SCOPE SYNC
9497
9498  033030  010035                2$:     MOV     R0,@(R5)+       ;TEST THE MOV
9499
9500  033032  020412                        CMP     R4,(R2)         ;CORRECT RESULT
9501  033034  001402                        BEQ     TST467          ;;BR IF YES
9502
9503  033036  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9504  033040  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9505
9506                                ;;**************************************************************
9507                                ;*TEST 467       MOV  SM0,DM4 TEST
9508                                ;;**************************************************************
9509  033042                        TST467:
9510  033042  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9511  033044  012700  000467                MOV     #467,R0         ;;LOAD R0 WITH TEST NUMBER
9512  033050  013701  033072                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9513  033054  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9514  033060  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9515  033062  012705  063314                MOV     #MBUF0+2,R5     ;R5 CONTAINS BASE DEST ADDR
9516  033066  005012                        CLR     (R2)            ;[DEST] = 000000
9517  033070  000257                        CCC                     ;SCOPE SYNC
9518
9519  033072  010045                2$:     MOV     R0,-(R5)        ;TEST THE MOV
9520
9521  033074  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
9522  033076  001402                        BEQ     TST470          ;;BR IF YES
9523
9524  033100  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9525  033102  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9526
9527                                ;;**************************************************************
9528                                ;*TEST 470       MOV  SM0,DM5 TEST
9529                                ;;**************************************************************
9530  033104                        TST470:
9531  033104  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9532  033106  012700  000470                MOV     #470,R0         ;;LOAD R0 WITH TEST NUMBER
9533  033112  013701  033134                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9534  033116  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9535  033122  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9536  033124  012705  063310                MOV     #ATA+12,R5      ;R5 CONTAINS BASE DEST ADDR
9537  033130  005012                        CLR     (R2)            ;[DEST] = 000000
9538  033132  000257                        CCC                     ;SCOPE SYNC
9539
9540  033134  010055                2$:     MOV     R0,@-(R5)       ;TEST THE MOV
9541
9542  033136  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
9543  033140  001402                        BEQ     TST471          ;;BR IF YES
9544
9545  033142  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9546  033144  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9547
9548
```

```
 9549                                   ;;*****************************************************************
 9550                                   ;*TEST 471      MOV  SM0,DM6 TEST
 9551                                   ;;*****************************************************************
 9552  033146                           ↑ST471:
 9553  033146  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9554  033150  012700  000471                  MOV     #471,R0         ;;LOAD R0 WITH TEST NUMBER
 9555  033154  013701  033176                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9556  033160  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 9557  033164  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9558  033166  012705  063312                  MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
 9559  033172  005012                          CLR     (R2)            ;[DEST] = 000000
 9560  033174  000257                          CCC                     ;SCOPE SYNC
 9561
 9562  033176  010065  000004          2$:     MOV     R0,4(R5)        ;TEST THE MOV
 9563
 9564  033202  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9565  033204  001402                          BEQ     ↑ST472          ;;BR IF YES
 9566
 9567  033206  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9568  033210  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9569
 9570                                   ;;*****************************************************************
 9571                                   ;*TEST 472      MOVB TEST - SM0,DM0 - EXTEND 1'S
 9572                                   ;;*****************************************************************
 9573  033212                           ↑ST472:
 9574  033212  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9575  033214  012700  000472                  MOV     #472,R0         ;;LOAD R0 WITH TEST NUMBER
 9576  033220  013701  033242                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9577  033224  012704  177652                  MOV     #177652,R4      ;RESULT S / B = 177652
 9578  033230  012705  000252                  MOV     #252,R5         ;SOURCE OP = 252
 9579  033234  005003                          CLR     R3              ;[DEST] = 000000
 9580  033236  000257                          CCC                     ;CLEAR FLAGS
 9581  033240  000266                          266                     ;N:C = 0110
 9582
 9583  033242  110503                  2$:     MOVB    R5,R3           ;TEST THE MOVB
 9584
 9585  033244  100003                          BPL     3$              ;N:C = 1000 ?
 9586  033246  001402                          BEQ     3$
 9587  033250  102401                          BVS     3$
 9588  033252  103001                          BCC     4$
 9589
 9590  033254  104002                  3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9591
 9592  033256  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9593  033260  001401                          BEQ     ↑ST473          ;;BR IF YES
 9594
 9595  033262  104002                  5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9596
 9597                                   ;;*****************************************************************
 9598                                   ;*TEST 473      MOVB TEST - SM0,DM0 - EXTEND 0'S
 9599                                   ;;*****************************************************************
 9600  033264                           ↑ST473:
 9601  033264  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9602  033266  012700  000473                  MOV     #473,R0         ;;LOAD R0 WITH TEST NUMBER
 9603  033272  013701  033314                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9604  033276  005004                          CLR     R4              ;RESULT S / B = 000000
```

```
 9605  033300  012705  177400          MOV     #177400,R5      ;SOURCE OP = 177400
 9606  033304  005003                  CLR     R3              ;[DEST] = 177777
 9607  033306  005103                  COM     R3
 9608  033310  000257                  CCC                     ;CLEAR FLAGS
 9609  033312  000271                  271                     ;N:C = 1001
 9610
 9611  033314  110503          2$:     MOVB    R5,R3           ;TEST THE MOVB
 9612
 9613  033316  100403                  BMI     3$              ;N:C = 0101 ?
 9614  033320  001002                  BNE     3$
 9615  033322  102401                  BVS     3$
 9616  033324  103401                  BCS     4$
 9617
 9618  033326  104002          3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9619
 9620  033330  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9621  033332  001401                  BEQ     TST474          ;;BR IF YES
 9622
 9623  033334  104002          5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9624
 9625                          ;;**********************************************************************
 9626                          ;*TEST 474       MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
 9627                          ;;**********************************************************************
 9628  033336                  TST474:
 9629  033336  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9630  033340  012700  000474          MOV     #474,R0         ;;LOAD R0 WITH TEST NUMBER
 9631  033344  013701  033364          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9632  033350  005004                  CLR     R4              ;RESULT S / B = 000000
 9633  033352  012705  064630          MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
 9634  033356  005003                  CLR     R3              ;[DEST] = 177777
 9635  033360  005103                  COM     R3
 9636  033362  000257                  CCC                     ;SCOPE SYNC
 9637
 9638  033364  111503          2$:     MOVB    (R5) R3         ;TEST THE MOVB
 9639
 9640  033366  020403                  CMP     R4,R3           ;RESULT CORRECT ?
 9641  033370  001401                  BEQ     TST475          ;;BR IF YES
 9642
 9643  033372  104002          3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9644
 9645                          ;;**********************************************************************
 9646                          ;*TEST 475       MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
 9647                          ;;**********************************************************************
 9648  033374                  TST475:
 9649  033374  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9650  033376  012700  000475          MOV     #475,R0         ;;LOAD R0 WITH TEST NUMBER
 9651  033402  013701  033424          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9652  033406  012704  000125          MOV     #125,R4         ;RESULT S / B = 125
 9653  033412  012705  064633          MOV     #DBTA+3,R5      ;SOURCE ADDR = DBTA+3
 9654  033416  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
 9655  033422  000257                  CCC                     ;SCOPE SYNC
 9656
 9657  033424  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
 9658
 9659  033426  020403                  CMP     R4,R3           ;RESULT CORRECT ?
 9660  033430  001401                  BEQ     TST476          ;;BR IF YES
```

```
9661
9662  033432  104002                  3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9663
9664                                  ;;*********************************************************************
9665                                  ;*TEST 476       MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
9666                                  ;;*********************************************************************
9667  033434                          TST476:
9668  033434  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9669  033436  012700  000476                  MOV     #476,R0         ;;LOAD R0 WITH TEST NUMBER
9670  033442  013701  033462                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9671  033446  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
9672  033452  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
9673  033456  005003                          CLR     R3              ;[DEST] = 000000
9674  033460  000257                          CCC                     ;SCOPE SYNC
9675
9676  033462  112503                  2$:     MOVB    (R5)+,R3        ;TEST THE MOVB
9677
9678  033464  020403                          CMP     R4,R3           ;RESULT CORRECT ?
9679  033466  001401                          BEQ     4$              ;BR IF YES
9680
9681  033470  104002                  3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9682
9683  033472  022705  064632          4$:     CMP     #DBTA+2,R5      ;DID MOVB INCREMENT SRC REG ?
9684  033476  001401                          BEQ     TST477          ;;BR IF YES
9685
9686  033500  104005                  5$:     ERROR   5               ;MOVB FAILED TO UPDATE SRC REG
9687
9688                                  ;;*********************************************************************
9689                                  ;*TEST 477       MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
9690                                  ;;*********************************************************************
9691  033502                          TST477:
9692  033502  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9693  033504  012700  000477                  MOV     #477,R0         ;;LOAD R0 WITH TEST NUMBER
9694  033510  013701  033530                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9695  033514  005004                          CLR     R4              ;RESULT S / B = 000000
9696  033516  012705  064630                  MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
9697  033522  012703  177400                  MOV     #177400,R3      ;[DEST] = 177400
9698  033526  000257                          CCC                     ;SCOPE SYNC
9699
9700  033530  112503                  2$:     MOVB    (R5)+,R3                ;TEST THE MOVB
9701
9702  033532  020403                          CMP     R4,R3           ;RESULT CORRECT ?
9703  033534  001401                          BEQ     4$              ;BR IF YES
9704
9705  033536  104002                  3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9706
9707  033540  022705  064631          4$:     CMP     #DBTA+1,R5      ;DID MOVB INCREMENT SRC REG ?
9708  033544  001401                          BEQ     TST500          ;;BR IF YES
9709
9710  033546  104005                  5$:     ERROR   5               ;MOVB FAILED TO UPDATE SOURCE REG
9711
9712
9713                                  ;;*********************************************************************
9714                                  ;*TEST 500       MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
9715                                  ;;*********************************************************************
9716  033550                          TST500:
9716  033550  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9717  033552  012700  000500                MOV     #500,R0         ;;LOAD R0 WITH TEST NUMBER
 9718  033556  013701  033602                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9719  033562  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9720  033566  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9721  033572  012705  064631                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9722  033576  005012                        CLR     (R2)            ;[DEST] = 000000
 9723  033600  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9724
 9725  033602  111512              2$:       MOVB    (R5),(R2)       ;TEST THE MOVB
 9726
 9727  033604  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
 9728  033606  001402                        BEQ     TST501          ;;BR IF YES
 9729
 9730  033610  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9731  033612  104001              3$:       ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9732
 9733                                        ;;*********************************************************************
 9734                                        ;*TEST 501      MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
 9735                                        ;;*********************************************************************
 9736  033614                                TST501:
 9737  033614  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9738  033616  012700  000501                MOV     #501,R0         ;;LOAD R0 WITH TEST NUMBER
 9739  033622  013701  033650                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9740  033626  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9741  033632  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9742  033636  012705  064631                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9743  033642  005012                        CLR     (R2)            ;[DEST] = 000000
 9744  033644  010203                        MOV     R2,R3           ;[R3] = DEST ADDR
 9745  033646  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9746
 9747  033650  111523              2$:       MOVB    (R5),(R3)+      ;TEST THE MOVB
 9748
 9749  033652  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
 9750  033654  001402                        BEQ     4$              ;BR IF YES
 9751
 9752  033656  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9753  033660  104001              3$:       ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9754
 9755  033662  022703  063313      4$:       CMP     #MBUF0+1,R3     ;DID MOVB INCREMENT THE DEST REG ?
 9756  033666  001401                        BEQ     TST502          ;;BR IF YES
 9757
 9758  033670  104005              5$:       ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9759
 9760                                        ;;*********************************************************************
 9761                                        ;*TEST 502      MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
 9762                                        ;;*********************************************************************
 9763  033672                                TST502:
 9764  033672  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9765  033674  012700  000502                MOV     #502,R0         ;;LOAD R0 WITH TEST NUMBER
 9766  033700  013701  033730                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9767  033704  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9768  033710  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9769  033714  012705  064631                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9770  033720  005012                        CLR     (R2)            ;[DEST] = 000000
 9771  033722  012703  063306                MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA +10
 9772  033726  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
```

```
 9773
 9774  033730  111533                 2$:     MOVB    (R5),@(R3)+     ;TEST THE MOVB
 9775
 9776  033732  022703  063310                 CMP     #ATA+12,R3      ;DID DEST REG GET INCREMENTED ?
 9777  033736  001401                         BEQ     4$              ;BR IF YES
 9778
 9779  033740  104005                 3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9780
 9781  033742  020412                 4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9782  033744  001402                         BEQ     TST503          ;;BR IF YES
 9783
 9784  033746  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 9785  033750  104001                 5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9786
 9787                                 ;;**************************************************************
 9788                                 ;*TEST 503      MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
 9789                                 ;;**************************************************************
 9790  033752                         TST503:
 9791  033752  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9792  033754  012700  000503                 MOV     #503,R0         ;;LOAD R0 WITH TEST NUMBER
 9793  033760  013701  034010                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9794  033764  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9795  033770  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377
 9796  033774  012705  064631                 MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9797  034000  005012                         CLR     (R2)            ;[DEST] = 000000
 9798  034002  012703  063313                 MOV     #MBUF0+1,R3     ;INITIAL DEST ADDR = MBUF0+1
 9799  034006  000257                         CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9800
 9801  034010  111543                 2$:     MOVB    (R5),-(R3)      ;TEST THE MOVB
 9802
 9803  034012  020302                         CMP     R3,R2           ;DID MOVB DECREMENT DEST REG ?
 9804  034014  001401                         BEQ     4$              ;BR IF YES
 9805
 9806  034016  104005                 3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9807
 9808  034020  020412                 4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9809  034022  001402                         BEQ     TST504          ;;BR IF YES
 9810
 9811  034024  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 9812  034026  104001                 5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9813
 9814                                 ;;**************************************************************
 9815                                 ;*TEST 504      MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
 9816                                 ;;**************************************************************
 9817  034030                         TST504:
 9818  034030  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9819  034032  012700  000504                 MOV     #504,R0         ;;LOAD R0 WITH TEST NUMBER
 9820  034036  013701  034066                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9821  034042  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9822  034046  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377
 9823  034052  012705  064631                 MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9824  034056  005012                         CLR     (R2)            ;[DEST] = 000000
 9825  034060  012703  063310                 MOV     #ATA+12,R3      ;INITIAL DEST ADDR = ATA +12
 9826  034064  000257                         CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9827
 9828  034066  111553                 2$:     MOVB    (R5),@-(R3)     ;TEST THE MOVB
```

```
 9829
 9830  034070  022703  063306                CMP     #ATA+10,R3      ;DID MOVB DECREMENT DEST REG ?
 9831  034074  001401                        BEQ     4$              ;BR IF YES
 9832
 9833  034076  104005            3$:         ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9834
 9835  034100  020412            4$:         CMP     R4,(R2)         ;CORRECT RESULT ?
 9836  034102  001402                        BEQ     TST505          ;;BR IF YES
 9837
 9838  034104  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9839  034106  104001            5$:         ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9840
 9841                            ;;*********************************************************************
 9842                            ;*TEST 505      MOVB TEST - SM1,DM6 - SRC ADR 000 / DST ADR EVEN
 9843                            ;;*********************************************************************
 9844  034110                    TST505:
 9845  034110  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9846  034112  012700  000505                MOV     #505,R0         ;;LOAD R0 WITH TEST NUMBER
 9847  034116  013701  034146                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9848  034122  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9849  034126  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9850  034132  012705  064631                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9851  034136  005012                        CLR     (R2)            ;[DEST] = 000000
 9852  034140  012703  063320                MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
 9853  034144  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9854
 9855  034146  111563  177772    2$:         MOVB    (R5),-6(R3)     ;TEST THE MOVB
 9856
 9857  034152  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
 9858  034154  001402                        BEQ     TST506          ;;BR IF YES
 9859
 9860  034156  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9861  034160  104001            3$:         ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9862
 9863                            ;;*********************************************************************
 9864                            ;*TEST 506      MOVB TEST - SM1,DM7 - SRC ADR 000 / DST ADR EVEN
 9865                            ;;*********************************************************************
 9866  034162                    TST506:
 9867  034162  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9868  034164  012700  000506                MOV     #506,R0         ;;LOAD R0 WITH TEST NUMBER
 9869  034170  013701  034220                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9870  034174  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9871  034200  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9872  034204  012705  064631                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9873  034210  005012                        CLR     (R2)            ;[DEST] = 000000
 9874  034212  012703  063276                MOV     #ATA,R3         ;BASE DEST ADDR = ATA
 9875  034216  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9876
 9877  034220  111573  000010    2$:         MOVB    (R5),@10(R3)    ;TEST THE MOVB
 9878
 9879  034224  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
 9880  034226  001402                        BEQ     TST507          ;;BR IF YES
 9881
 9882  034230  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9883  034232  104001            3$:         ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9884
```

```
                                ;;*****************************************************************
                                ;*TEST 507       MOVB  SM0,DM1 TEST
                                ;;*****************************************************************
034234                          TST507:
034234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
034236  012700  000507                  MOV     #507,R0         ;;LOAD R0 WITH TEST NUMBER
034242  013701  034270                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
034246  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
034252  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
034256  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
034262  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
034264  005012                          CLR     (R2)            ;[DEST] = 000000
034266  000257                          CCC                     ;SCOPE SYNC

034270  110315                  2$:     MOVB    R3,(R5)         ;TEST THE MOVB

034272  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
034274  001402                          BEQ     TST510          ;;BR IF YES

034276  011203                          MOV     (R2),R3         ;GET THE WAS DATA
034300  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 510       MOVB  SM0,DM2 TEST
                                ;;*****************************************************************
034302                          TST510:
034302  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
034304  012700  000510                  MOV     #510,R0         ;;LOAD R0 WITH TEST NUMBER
034310  013701  034336                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
034314  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
034320  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
034324  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
034330  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
034332  005012                          CLR     (R2)            ;[DEST] = 000000
034334  000257                          CCC                     ;SCOPE SYNC

034336  110325                  2$:     MOVB    R3,(R5)+        ;TEST THE MOVB

034340  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
034342  001402                          BEQ     TST511          ;;BR IF YES

034344  011203                          MOV     (R2),R3         ;GET THE WAS DATA
034346  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 511       MOVB  SM0,DM3 TEST
                                ;;*****************************************************************
034350                          TST511:
034350  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
034352  012700  000511                  MOV     #511,R0         ;;LOAD R0 WITH TEST NUMBER
034356  013701  034406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
034362  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
034366  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
034372  012703  177777                  MOV     #-1,R3          ;SOURCE OP IN R3
034376  012705  063306                  MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
034402  005012                          CLR     (R2)            ;[DEST] = 000000
```

```
9941  034404  000257                          CCC                     ;SCOPE SYNC
9942
9943  034406  110335                  2$:     MOVB    R3,@(R5)+       ;TEST THE MOVB
9944
9945  034410  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9946  034412  001402                          BEQ     TST512          ;;BR IF YES
9947
9948  034414  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9949  034416  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
9950
9951                                  ;;**************************************************************
9952                                  ;*TEST 512       MOVB  SM0,DM4 TEST
9953                                  ;;**************************************************************
9954  034420                          TST512:
9955  034420  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9956  034422  012700  000512                  MOV     #512,R0         ;;LOAD R0 WITH TEST NUMBER
9957  034426  013701  034456                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9958  034432  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9959  034436  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
9960  034442  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
9961  034446  012705  063314                  MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
9962  034452  005012                          CLR     (R2)            ;[DEST] = 000000
9963  034454  000257                          CCC                     ;SCOPE SYNC
9964
9965  034456  110345                  2$:     MOVB    R3,-(R5)        ;TEST THE MOVB
9966
9967  034460  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9968  034462  001402                          BEQ     TST513          ;;BR IF YES
9969
9970  034464  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9971  034466  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
9972
9973                                  ;;**************************************************************
9974                                  ;*TEST 513       MOVB  SM0,DM6 TEST
9975                                  ;;**************************************************************
9976  034470                          TST513:
9977  034470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9978  034472  012700  000513                  MOV     #513,R0         ;;LOAD R0 WITH TEST NUMBER
9979  034476  013701  034526                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9980  034502  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9981  034506  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
9982  034512  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
9983  034516  012705  063314                  MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
9984  034522  005012                          CLR     (R2)            ;[DEST] = 000000
9985  034524  000257                          CCC                     ;SCOPE SYNC
9986
9987  034526  110365  177776          2$:     MOVB    R3,-2(R5)       ;TEST THE MOVB
9988
9989  034532  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9990  034534  001402                          BEQ     TST514          ;;BR IF YES
9991
9992  034536  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9993  034540  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
9994
9995                                  ;;**************************************************************
9996                                  ;*TEST 514       BIS TEST - SM0,DM0 - N:C = 0111
```

```
 9997
 9998  034542                          ;;*****************************************************************
 9999  034542                          TST514:
       034542  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10000  034544  012700  000514                  MOV     #514,R0         ;;LOAD R0 WITH TEST NUMBER
10001  034550  013701  034574                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10002  034554  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10003  034560  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
10004  034564  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
10005  034570  000257                          CCC                     ;CLEAR FLAGS
10006  034572  000267                          267                     ;N:C = 0111
10007
10008  034574  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10009
10010  034576  100003                          BPL     3$              ;N:C = 1001 ?
10011  034600  001402                          BEQ     3$
10012  034602  102401                          BVS     3$
10013  034604  103401                          BCS     4$
10014
10015  034606  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10016
10017  034610  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10018  034612  001401                          BEQ     TST515          ;;BR IF YES
10019
10020  034614  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10021
10022                                  ;;*****************************************************************
10023                                  ;TEST 515       BIS TEST - SM0,DM0 - N:C = 1000
10024                                  ;;*****************************************************************
10025  034616                          TST515:
10026  034616  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10027  034620  012700  000515                  MOV     #515,R0         ;;LOAD R0 WITH TEST NUMBER
10028  034624  013701  034654                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10029                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT10
10030  034630  032737  002000  063234          BIT     #BIT10,@#BPTLOC ;BREAKPOINT HALT SET ??
10031  034636  001401                          BEQ     .+4             ;BR IF NOT
10032  034640  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10033  034642  005004                          CLR     R4              ;RESULT S / B = 000000
10034  034644  005005                          CLR     R5              ;SRC OPR = 000000
10035  034646  005003                          CLR     R3              ;[DEST] = 000000
10036  034650  000257                          CCC                     ;CLEAR FLAGS
10037  034652  000270                          SEN                     ;N:C = 1000
10038
10039  034654  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10040
10041  034656  100403                          BMI     3$              ;N:C = 0100
10042  034660  001002                          BNE     3$
10043  034662  102401                          BVS     3$
10044  034664  103001                          BCC     4$
10045
10046  034666  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10047
10048  034670  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10049  034672  001401                          BEQ     TST516          ;;BR IF YES
10050
10051  034674  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10052
```

```
10053                                   ;;*****************************************************************
10054                                   ;*TEST 516      BIC TEST - SM0,DM0 - N:C = 0111
10055                                   ;;*************  *************************************************
10056  034676                           TST516:
10057  034676  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10058  034700  012700  000516                   MOV     #516,R0         ;;LOAD R0 WITH TEST NUMBER
10059  034704  013701  034730                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10060  034710  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
10061  034714  012705  077777                   MOV     #77777,R5       ;SRC OPR = 77777
10062  034720  012703  177777                   MOV     #-1,R3          ;[DEST] = 177777
10063  034724  000257                           CCC                     ;CLEAR FLAGS
10064  034726  000267                           SEV                     ;N:C = 0111
10065
10066  034730  040503                   2$:     BIC     R5,R3           ;TEST THE BIC
10067
10068  034732  100003                           BPL     3$              ;N:C = 1001 ?
10069  034734  001402                           BEQ     3$
10070  034736  102401                           BVS     3$
10071  034740  103401                           BCS     4$
10072
10073  034742  104002                   3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10074
10075  034744  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10076  034746  001401                           BEQ     TST517          ;;BR IF YES
10077
10078  034750  104002                   5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10079
10080                                   ;;*****************************************************************
10081                                   ;*TEST 517      BIC TEST - SM0,DM0 - N:C = 1000
10082                                   ;;*****************************************************************
10083  034752                           TST517:
10084  034752  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10085  034754  012700  000517                   MOV     #517,R0         ;;LOAD R0 WITH TEST NUMBER
10086  034760  013701  034776                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10087  034764  005004                           CLR     R4              ;RESULT S / B = 000000
10088  034766  005005                           CLR     R5              ;SRC OPR = 000000
10089  034770  005003                           CLR     R3              ;[DEST] = 000000
10090  034772  000257                           CCC                     ;CLEAR FLAGS
10091  034774  000270                           SEN                     ;N:C = 1000
10092
10093  034776  040503                   2$:     BIC     R5,R3           ;TEST THE BIC
10094
10095  035000  100403                           BMI     3$              ;N:C = 0100
10096  035002  001002                           BNE     3$
10097  035004  102401                           BVS     3$
10098  035006  103001                           BCC     4$
10099
10100  035010  104002                   3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10101
10102  035012  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10103  035014  001401                           BEQ     TST520          ;;BR IF YES
10104
10105  035016  104002                   5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10106
10107                                   ;;*****************************************************************
10108                                   ;*TEST 520      BIT TEST - SM0,DM0 - N:C = 0111
```

```
10109                                   ;;*****************************************************************
10110  035020                           T$T520:
10111  035020  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10112  035022  012700  000520                   MOV     #520,R0         ;;LOAD R0 WITH TEST NUMBER
10113  035026  013701  035052                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10114  035032  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
10115  035036  012705  100000                   MOV     #100000,R5      ;SRC OPR = 100000
10116  035042  012703  100000                   MOV     #100000,R3      ;[DEST] = 100000
10117  035046  000257                           CCC                     ;CLEAR FLAGS
10118  035050  000267                           267                     ;N:C = 0111
10119
10120  035052  030503                   2$:     BIT     R5,R3           ;TEST THE BIT
10121
10122  035054  100003                           BPL     3$              ;N:C = 1001
10123  035056  001402                           BEQ     3$
10124  035060  102401                           BVS     3$
10125  035062  103401                           BCS     4$
10126
10127  035064  104002                   3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10128
10129  035066  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10130  035070  001402                           BEQ     T$T521          ;;BR IF YES
10131
10132  035072  011203                           MOV     (R2),R3         ;GET THE WAS DATA
10133  035074  104002                   5$:     ERROR   2               ;BIT DELIVERED A RESULT
10134
10135                                   ;;*****************************************************************
10136                                   ;T$TEST 521     BIT TEST - SMD,DMD - N:C = 1000
10137                                   ;;*****************************************************************
10138  035076                           T$T521:
10139  035076  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10140  035100  012700  000521                   MOV     #521,R0         ;;LOAD R0 WITH TEST NUMBER
10141  035104  013701  035126                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10142  035110  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
10143  035114  012705  052525                   MOV     #52525,R5       ;SRC OPR = 52525
10144  035120  010403                           MOV     R4,R3           ;[DEST] = 125252
10145  035122  000257                           CCC                     ;CLEAR FLAGS
10146  035124  000270                           SEN                     ;N:C = 1000
10147
10148  035126  030503                   2$:     BIT     R5,R3   ;TEST THE BIT
10149
10150  035130  100403                           BMI     3$              ;N:C = 0100
10151  035132  001002                           BNE     3$
10152  035134  102401                           BVS     3$
10153  035136  103001                           BCC     4$
10154
10155  035140  104002                   3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10156
10157  035142  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10158  035144  001401                           BEQ     T$T522          ;;BR IF YES
10159
10160  035146  104002                   5$:     ERROR   2               ;BIT DELIVERED A RESULT
10161
10162                                   ;;*****************************************************************
10163                                   ;T$TEST 522     CMP TEST - SMD,DMD - N:C = 0110
10164                                   ;;*****************************************************************
```

```
10165  035150                          TST522:
10166  035150  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10167  035152  012700  000522                  MOV     #522,R0         ;;LOAD R0 WITH TEST NUMBER
10168  035156  013701  035200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10169  035162  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10170  035166  005005                          CLR     R5              ;SRC OPR = 000000
10171  035170  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10172  035174  000257                          CCC                     ;CLEAR FLAGS
10173  035176  000266                          266                     ;N:C = 0110
10174
10175  035200  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
10176
10177  035202  100003                          BPL     3$              ;N:C = 1001
10178  035204  001402                          BEQ     3$
10179  035206  102401                          BVS     3$
10180  035210  103401                          BCS     4$
10181
10182  035212  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10183
10184  035214  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10185  035216  001401                          BEQ     TST523          ;;BR IF YES
10186
10187  035220  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10188
10189                                  ;;**************************************************************
10190                                  ;*TEST 523      CMP TEST - SM0,DM0 - N:C = 1010
10191                                  ;;**************************************************************
10192  035222                          TST523:
10193  035222  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10194  035224  012700  000523                  MOV     #523,R0         ;;LOAD R0 WITH TEST NUMBER
10195  035230  013701  035252                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10196  035234  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10197  035240  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
10198  035244  010403                          MOV     R4,R3           ;[DEST] = 177777
10199  035246  000257                          CCC                     ;CLEAR FLAGS
10200  035250  000272                          272                     ;N:C = 1010
10201
10202  035252  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
10203
10204  035254  100403                          BMI     3$              ;N:C = 0100
10205  035256  001002                          BNE     3$
10206  035260  102401                          BVS     3$
10207  035262  103001                          BCC     4$
10208
10209  035264  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10210
10211  035266  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10212  035270  001401                          BEQ     TST524          ;;BR IF YES
10213
10214  035272  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10215
10216                                  ;;**************************************************************
10217                                  ;*TEST 524      CMP TEST - SM0,DM0 - N:C = 0000
10218                                  ;;**************************************************************
10219  035274                          TST524:
10220  035274  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
10221  035276  012700  000524          MOV     #524,R0         ;;LOAD R0 WITH TEST NUMBER
10222  035302  013701  035324          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10223  035306  012704  000001          MOV     #+1,R4          ;RESULT S / B = +1
10224  035312  012705  100000          MOV     #100000,R5                      ;SRC OPR = 100000
10225  035316  012703  000001          MOV     #+1,R3          ;[DEST] = +1
10226  035322  000257                  CCC                     ;CLEAR FLAGS
10227
10228  035324  020503          2$:     CMP     R5,R3           ;TEST THE CMP
10229
10230  035326  100403                  BMI     3$              ;N:C = 0010
10231  035330  001402                  BEQ     3$
10232  035332  102001                  BVC     3$
10233  035334  103001                  BCC     4$
10234
10235  035336  104002          3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10236
10237  035340  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10238  035342  001401                  BEQ     TST525          ;;BR IF YES
10239
10240  035344  104002          5$:     ERROR   2               ;CMP DELIVERED A RESULT
10241
10242                          ;;**************************************************************
10243                          ;*TEST 525      BIS TEST - SM0,DM1 - N:C = 0111
10244                          ;;**************************************************************
10245  035346                  TST525:
10246  035346  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10247  035350  012700  000525          MOV     #525,R0         ;;LOAD R0 WITH TEST NUMBER
10248  035354  013701  035404          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10249  035360  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10250  035364  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
10251  035370  012705  125252          MOV     #125252,R5      ;SRC OPR = 125252
10252  035374  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
10253  035400  000257                  CCC                     ;CLEAR FLAGS
10254  035402  000267                  267                     ;N:C = 0111
10255
10256  035404  050512          2$:     BIS     R5,(R2)         ;TEST THE BIS
10257
10258  035406  100003                  BPL     3$              ;N:C = 1001
10259  035410  001402                  BEQ     3$
10260  035412  102401                  BVS     3$
10261  035414  103401                  BCS     4$
10262
10263  035416  104001          3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10264
10265  035420  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10266  035422  001402                  BEQ     TST526          ;;BR IF YES
10267
10268  035424  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10269  035426  104001          5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10270
10271                          ;;**************************************************************
10272                          ;*TEST 526      BIS TEST - SM0,DM1 - N:C = 1000
10273                          ;;**************************************************************
10274  035430                  TST526:
10275  035430  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10276  035432  012700  000526          MOV     #526,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
10277  035436  013701  035460          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10278  035442  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10279  035446  005004                  CLR     R4              ;RESULT S / B = 000000
10280  035450  005005                  CLR     R5              ;SRC OPR = 000000
10281  035452  005012                  CLR     (R2)            ;[DEST] = 000000
10282  035454  000257                  CCC                     ;CLEAR FLAGS
10283  035456  000270                  SEN                     ;N:C = 1000
10284
10285  035460  050512          2$:     BIS     R5,(R2)         ;TEST THE BIS
10286
10287  035462  100403                  BMI     3$              ;N:C = 0100
10288  035464  001002                  BNE     3$
10289  035466  102401                  BVS     3$
10290  035470  103001                  BCC     4$
10291
10292  035472  104001          3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10293
10294  035474  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10295  035476  001402                  BEQ     TST527          ;;BR IF YES
10296
10297  035500  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10298  035502  104001          5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10299
10300                          ;;**************************************************************
10301                          ;*TEST 527      BIC TEST - SM0,DM1 - N:C = 0111
10302                          ;;**************************************************************
10303  035504                  TST527:
10304  035504  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10305  035506  012700  000527          MOV     #527,R0         ;;LOAD R0 WITH TEST NUMBER
10306  035512  013701  035542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10307  035516  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10308  035522  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
10309  035526  012705  077777          MOV     #77777,R5               ;SRC OPR = 77777
10310  035532  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
10311  035536  000257                  CCC                     ;CLEAR FLAGS
10312  035540  000267                  267                     ;N:C = 0111
10313
10314  035542  040512          2$:     BIC     R5,(R2)         ;TEST THE BIC
10315
10316  035544  100003                  BPL     3$              ;N:C = 1001
10317  035546  001402                  BEQ     3$
10318  035550  102401                  BVS     3$
10319  035552  103401                  BCS     4$
10320
10321  035554  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10322
10323  035556  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10324  035560  001402                  BEQ     TST530          ;;BR IF YES
10325
10326  035562  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10327  035564  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10328
10329                          ;;**************************************************************
10330                          ;*TEST 530      BIC TEST - SM0,DM1 - N:C = 1000
10331                          ;;**************************************************************
10332  035566                  TST530:
```

```
10333  035566  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10334  035570  012700  000530                  MOV     #530,R0         ;;LOAD R0 WITH TEST NUMBER
10335  035574  013701  035616                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10336  035600  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10337  035604  005004                          CLR     R4              ;RESULT S / B = 000000
10338  035606  005005                          CLR     R5              ;SRC OPR = 000000
10339  035610  005012                          CLR     (R2)            ;[DEST] = 000000
10340  035612  000257                          CCC                     ;CLEAR FLAGS
10341  035614  000270                          SEN                     ;N:C = 1000
10342
10343  035616  040512                  2$:     BIC     R5,(R2)         ;TEST THE BIC
10344
10345  035620  100403                          BMI     3$              ;N:C = 0100
10346  035622  001002                          BNE     3$
10347  035624  102401                          BVS     3$
10348  035626  103001                          BCC     4$
10349
10350  035630  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10351
10352  035632  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10353  035634  001402                          BEQ     TST531          ;;BR IF YES
10354
10355  035636  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10356  035640  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10357
10358                                  ;;**********************************************************************
10359                                  ;*TEST 531      BIT TEST - SM0,DM1 - N:C = 0111
10360                                  ;;**********************************************************************
10361  035642                          TST531:
10362  035642  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10363  035644  012700  000531                  MOV     #531,R0         ;;LOAD R0 WITH TEST NUMBER
10364  035650  013701  035700                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10365  035654  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10366  035660  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10367  035664  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10368  035670  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10369  035674  000257                          CCC                     ;CLEAR FLAGS
10370  035676  000267                          267                     ;N:C = 0111
10371
10372  035700  030512                  2$:     BIT     R5,(R2)         ;TEST THE BIT
10373
10374  035702  100003                          BPL     3$              ;N:C = 1001
10375  035704  001402                          BEQ     3$
10376  035706  102401                          BVS     3$
10377  035710  103401                          BCS     4$
10378
10379  035712  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10380
10381  035714  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10382  035716  001402                          BEQ     TST532          ;;BR IF YES
10383
10384  035720  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10385  035722  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10386
10387                                  ;;**********************************************************************
10388                                  ;*TEST 532      BIT TEST - SM0,DM1 - N:C = 1000
```

```
10389                                   ;;*****************************************************************
10390  035724                           TST532:
10391  035724  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10392  035726  012700  000532                   MOV     #532,R0         ;;LOAD R0 WITH TEST NUMBER
10393  035732  013701  035762                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10394  035736  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10395  035742  012704  052525                   MOV     #52525,R4       ;RESULT S / B = 52525
10396  035746  012705  125252                   MOV     #125252,R5      ;SRC OPR = 125252
10397  035752  012712  052525                   MOV     #52525,(R2)     ;[DEST] = 52525
10398  035756  000257                           CCC                     ;CLEAR FLAGS
10399  035760  000270                           SEN                     ;N:C = 1000
10400
10401  035762  030512                   2$:     BIT     R5,(R2)         ;TEST THE BIT
10402
10403  035764  100403                           BMI     3$              ;N:C = 0100
10404  035766  001002                           BNE     3$
10405  035770  102401                           BVS     3$
10406  035772  103001                           BCC     4$
10407
10408  035774  104001                   3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10409
10410  035776  020412                   4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10411  036000  001402                           BEQ     TST533          ;;BR IF YES
10412
10413  036002  011203                           MOV     (R2),R3         ;GET THE WAS DATA
10414  036004  104001                   5$:     ERROR   1               ;BIT DELIVERED A RESULT
10415                                   ;;*****************************************************************
10416                                   ;*TEST 533      CMP TEST - SM0,DM1 - N:C = 1010
10417                                   ;;*****************************************************************
10418  036006                           TST533:
10419  036006  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10420  036010  012700  000533                   MOV     #533,R0         ;;LOAD R0 WITH TEST NUMBER
10421  036014  013701  036044                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10422  036020  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10423  036024  012704  177777                   MOV     #-1,R4          ;RESULT S / B = -1
10424  036030  012705  177777                   MOV     #-1,R5          ;SRC OPR = 177777
10425  036034  012712  177777                   MOV     #-1,(R2)        ;[DEST] = 177777
10426  036040  000257                           CCC                     ;CLEAR FLAGS
10427  036042  000272                           272                     ;N:C = 1010
10428
10429  036044  020512                   2$:     CMP     R5,(R2)         ;TEST THE CMP
10430
10431  036046  100403                           BMI     3$              ;N:C = 0100
10432  036050  001002                           BNE     3$
10433  036052  102401                           BVS     3$
10434  036054  103001                           BCC     4$
10435
10436  036056  104001                   3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10437
10438  036060  020412                   4$:     CMP     R4,(F2)         ;CORRECT RESULT ?
10439  036062  001402                           BEQ     TST534          ;;BR IF YES
10440
10441  036064  011203                           MOV     (R2),R3         ;GET THE WAS DATA
10442  036066  104001                   5$:     ERROR   1               ;CMP DELIVERED A RESULT
10443
10444                                   ;;*****************************************************************
```

```
10445                                          ;*TEST 534      CMP TEST - SM0,DM1 - N:C = 0110
10446                                          ;;**************************************************************
10447  036070                                  TST534:
10448  036070  000004                                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
10449  036072  012700  000534                        MOV     #534,R0                 ;;LOAD R0 WITH TEST NUMBER
10450  036076  013701  036124                        MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
10451  036102  012702  063312                        MOV     #MBUF0,R2               ;DEST ADDR = MBUF0
10452  036106  012704  000001                        MOV     #+1,R4                  ;RESULT S / B = +1
10453  036112  005005                                CLR     R5                      ;SRC OPR = 000000
10454  036114  012712  000001                        MOV     #+1,(R2)                ;(DEST) = +1
10455  036120  000257                                 CCC                             ;CLEAR FLAGS
10456  036122  000266                                 266                             ;N:C = 0110
10457
10458  036124  020512                  2$:    CMP     R5,(R2)                 ;TEST THE CMP
10459
10460  036126  100003                         BPL     3$                      ;N:C = 1001
10461  036130  001402                         BEQ     3$
10462  036132  102401                         BVS     3$
10463  036134  103401                         BCS     4$
10464
10465  036136  104001                  3$:    ERROR   1                       ;CMP FAILED TO ALTER CODES PROPERLY
10466
10467  036140  020412                  4$:    CMP     R4,(R2)                 ;CORRECT RESULT ?
10468  036142  001402                         BEQ     TST535                  ;;BR IF YES
10469
10470  036144  011203                         MOV     (R2),R3                 ;GET THE WAS DATA
10471  036146  104001                  5$:    ERROR   1                       ;CMP DELIVERED A RESULT
10472
10473                                          ;;**************************************************************
10474                                          ;*TEST 535      CMP TEST - SM0,DM1 - N:C = 0000
10475                                          ;;**************************************************************
10476  036150                                  TST535:
10477  036150  000004                                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
10478  036152  012700  000535                        MOV     #535,R0                 ;;LOAD R0 WITH TEST NUMBER
10479  036156  013701  036204                        MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
10480  036162  012702  063312                        MOV     #MBUF0,R2               ;DEST ADDR = MBUF0
10481  036166  012704  000001                        MOV     #+1,R4                  ;RESULT S / B = +1
10482  036172  012705  100000                        MOV     #100000,R5              ;SRC OPR = 100000
10483  036176  012712  000001                        MOV     #+1,(R2)                ;(DEST) = +1
10484  036202  000257                                 CCC                             ;CLEAR FLAGS
10485
10486  036204  020512                  2$:    CMP     R5,(R2)                 ;TEST THE CMP
10487
10488  036206  100403                         BMI     3$                      ;N:C = 0010
10489  036210  001402                         BEQ     3$
10490  036212  102001                         BVC     3$
10491  036214  103001                         BCC     4$
10492
10493  036216  104001                  3$:    ERROR   1                       ;CMP FAILED TO ALTER CODES PROPERLY
10494
10495  036220  020412                  4$:    CMP     R4,(R2)                 ;CORRECT RESULT ?
10496  036222  001402                         BEQ     TST536                  ;;BR IF YES
10497
10498  036224  011203                         MOV     (R2),R3                 ;GET THE WAS DATA
10499  036226  104001                  5$:    ERROR   1                       ;CMP DELIVERED A RESULT
10500
```

```
10501                                   ;;*****************************************************************
10502                                   ;*TEST 536      BIS TEST - SM1,DM0 - N:C = 0111
10503                                   ;;*****************************************************************
10504  036230                           TST536:
10505  036230  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10506  036232  012700  000536                   MOV     #536,R0         ;;LOAD R0 WITH TEST NUMBER
10507  036236  013701  036262                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10508  036242  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
10509  036246  012705  063332                   MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10510  036252  012703  052525                   MOV     #52525,R3       ;[DEST] = 52525
10511  036256  000257                           CCC                     ;CLEAR FLAGS
10512  036260  000267                           267                     ;N:C = 0111
10513
10514  036262  051503                   2$:     BIS     (R5),R3         ;TEST THE BIS
10515
10516  036264  100003                           BPL     3$              ;N:C = 1001
10517  036266  001402                           BEQ     3$
10518  036270  102401                           BVS     3$
10519  036272  103401                           BCS     4$
10520
10521  036274  104002                   3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10522
10523  036276  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10524  036300  001401                           BEQ     TST537          ;;BR IF YES
10525
10526  036302  104002                   5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10527
10528                                   ;;*****************************************************************
10529                                   ;*TEST 537      BIS TEST - SM1,DM0 - N:C = 1000
10530                                   ;;*****************************************************************
10531  036304                           TST537:
10532  036304  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10533  036306  012700  000537                   MOV     #537,R0         ;;LOAD R0 WITH TEST NUMBER
10534  036312  013701  036332                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10535  036316  005004                           CLR     R4              ;RESULT S / B =000000
10536  036320  012705  063322                   MOV     #DWTA,R5        ;SRC ADDR = DWTA
10537  036324  005003                           CLR     R3              ;[DEST] = 000000
10538  036326  000257                           CCC                     ;CLEAR FLAGS
10539  036330  000270                           SEN                     ;N:C = 1000
10540
10541  036332  051503                   2$:     BIS     (R5),R3         ;TEST THE BIS
10542
10543  036334  100403                           BMI     3$              ;N:C = 0100
10544  036336  001002                           BNE     3$
10545  036340  102401                           BVS     3$
10546  036342  103001                           BCC     4$
10547
10548  036344  104002                   3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10549
10550  036346  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10551  036350  001401                           BEQ     TST540          ;;BR IF YES
10552
10553  036352  104002                   5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10554
10555                                   ;;*****************************************************************
10556                                   ;*TEST 540      BIC TEST - SM1,DM0 - N:C = 0111
```

```
10557                                   ;;*****************************************************************
10558  036354                           TST540:
10559  036354  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10560  036356  012700  000540                   MOV     #540,R0         ;;LOAD R0 WITH TEST NUMBER
10561  036362  013701  036412                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10562  036366  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
10563  036372  012705  063316                   MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10564  036376  012703  177777                   MOV     #-1,R3  ;[DEST] = 177777
10565  036402  012715  077777                   MOV     #77777,(R5)     ;SRC OPR = 77777
10566  036406  000257                           CCC                     ;CLEAR FLAGS
10567  036410  000267                           SCC                     ;N:C = 0111
10568
10569  036412  041503                   2$:     BIC     (R5),R3         ;TEST THE BIC
10570
10571  036414  100003                           BPL     3$              ;N:C = 1001 ?
10572  036416  001402                           BEQ     3$
10573  036420  102401                           BVS     3$
10574  036422  103401                           BCS     4$
10575
10576  036424  104002                   3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10577
10578  036426  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10579  036430  001401                           BEQ     TST541          ;;BR IF YES
10580
10581  036432  104002                   5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10582
10583                                   ;;*****************************************************************
10584                                   ;*TEST 541      BIC TEST - SM1,DM0 - N:C = 1000
10585                                   ;;*****************************************************************
10586  036434                           TST541:
10587  036434  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10588  036436  012700  000541                   MOV     #541,R0         ;;LOAD R0 WITH TEST NUMBER
10589  036442  013701  036462                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10590  036446  005004                           CLR     R4              ;RESULT S / B = 000000
10591  036450  012705  063322                   MOV     #DWTA,R5        ;SRC ADDR = DWTA
10592  036454  005003                           CLR     R3              ;[DEST] =000000
10593  036456  000257                           CCC                     ;CLEAR FLAGS
10594  036460  000270                           SEN                     ;N:C = 1000
10595
10596  036462  041503                   2$:     BIC     (R5),R3         ;TEST THE BIC
10597
10598  036464  100403                           BMI     3$              ;N:C = 0100
10599  036466  001002                           BNE     3$
10600  036470  102401                           BVS     3$
10601  036472  103001                           BCC     4$
10602
10603  036474  104002                   3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10604
10605  036476  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10606  036500  001401                           BEQ     TST542          ;;BR IF YES
10607
10608  036502  104002                   5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10609
10610                                   ;;*****************************************************************
10611                                   ;*TEST 542      BIT TEST - SM1,DM0 - N:C = 0111
10612                                   ;;*****************************************************************
```

```
10613  036504                       TST542:
10614  036504  000004                       SCOPE                    ;CALL THE SCOPE LOOP UTILITY
10615  036506  012700  000542               MOV     #542,R0          ;;LOAD R0 WITH TEST NUMBER
10616  036512  013701  036534               MOV     @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
10617  036516  012704  100000               MOV     #100000,R4       ;RESULT S / B = 100000
10618  036522  012705  063324               MOV     #DWTA+2,R5       ;SRC ADDR = DWTA+2
10619  036526  010403                       MOV     R4,R3            ;[DEST] = 100000
10620  036530  000257                       CCC                      ;CLEAR FLAGS
10621  036532  000267                       SCC                      ;N:C = 0111
10622
10623  036534  031503               2$:     BIT     (R5),R3          ;TEST THE BIT
10624
10625  036536  100003                       BPL     3$               ;N:C = 1001 ?
10626  036540  001402                       BEQ     3$
10627  036542  102401                       BVS     3$
10628  036544  103401                       BCS     4$
10629
10630  036546  104002               3$:     ERROR   2                ;BIT FAILED TO ALTER CODES PROPERLY
10631
10632  036550  020403               4$:     CMP     R4,R3            ;CORRECT RESULT ?
10633  036552  001401                       BEQ     TST543           ;;BR IF YES
10634
10635  036554  104002               5$:     ERROR   2                ;BIT DELIVERED A RESULT
10636
10637                                ;;**************************************************************
10638                                ;*TEST 543      BIT TEST - SM1,DM0 - N:C = 1000
10639                                ;;**************************************************************
10640  036556                       TST543:
10641  036556  000004                       SCOPE                    ;CALL THE SCOPE LOOP UTILITY
10642  036560  012700  000543               MOV     #543,R0          ;;LOAD R0 WITH TEST NUMBER
10643  036564  013701  036606               MOV     @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
10644  036570  012704  052525               MOV     #52525,R4        ;RESULT S / B = 52525
10645  036574  012705  063332               MOV     #DWTA+10,R5      ;SRC ADDR = DWTA+10
10646  036600  010403                       MOV     R4,R3            ;[DEST] = 52525
10647  036602  000257                       CCC                      ;CLEAR FLAGS
10648  036604  000270                       SEN                      ;N:C = 1000
10649
10650  036606  031503               2$:     BIT     (R5),R3          ;TEST THE BIT
10651
10652  036610  100403                       BMI     3$               ;N:C = 0100
10653  036612  001002                       BNE     3$
10654  036614  102401                       BVS     3$
10655  036616  103001                       BCC     4$
10656
10657  036620  104002               3$:     ERROR   2                ;BIT FAILED TO ALTER CODES PROPERLY
10658
10659  036622  020403               4$:     CMP     R4,R3            ;CORRECT RESULT ?
10660  036624  001401                       BEQ     TST544           ;;BR IF YES
10661
10662  036626  104002               5$:     ERROR   2                ;BIT DELIVERED A RESULT
10663                                ;;**************************************************************
10664                                ;*TEST 544      CMP TEST - SM1,DM0 - N:C = 0110
10665                                ;;**************************************************************
10666  036630                       TST544:
10667  036630  000004                       SCOPE                    ;CALL THE SCOPE LOOP UTILITY
10668  036632  012700  000544               MOV     #544,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
10669  036636  013701  036660          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10670  036642  012704  000001          MOV     #+1,R4          ;RESULT S / B = +1
10671  036646  012705  063322          MOV     #DWTA,R5        ;SRC ADDR = DWTA
10672  036652  010403                  MOV     R4,R3           ;[DEST] = +1
10673  036654  000257                  CCC                     ;CLEAR FLAGS
10674  036656  000266                  266                     ;N:C = 0110
10675
10676  036660  021503          2$:     CMP     (R5),R3         ;TEST THE CMP
10677
10678  036662  100003                  BPL     3$              ;N:C = 1001
10679  036664  001402                  BEQ     3$
10680  036666  102401                  BVS     3$
10681  036670  103401                  BCS     4$
10682
10683  036672  104002          3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10684
10685  036674  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10686  036676  001401                  BEQ     TST545          ;;BR IF YES
10687
10688  036700  104002          5$:     ERROR   2               ;CMP DELIVERED A RESULT
10689
10690                          ;;**********************************************************
10691                          ;*TEST 545      CMP TEST - SM1,DM0 - N:C = 1010
10692                          ;;**********************************************************
10693  036702                  TST545:
10694  036702  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10695  036704  012700  000545          MOV     #545,R0         ;;LOAD R0 WITH TEST NUMBER
10696  036710  013701  036732          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10697  036714  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
10698  036720  012705  063324          MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10699  036724  010403                  MOV     R4,R3           ;[DEST] = 177777
10700  036726  000257                  CCC                     ;CLEAR FLAGS
10701  036730  000272                  272                     ;N:C = 1010
10702
10703  036732  021503          2$:     CMP     (R5),R3         ;TEST THE CMP
10704
10705  036734  100403                  BMI     3$              ;N:C = 0100
10706  036736  001002                  BNE     3$
10707  036740  102401                  BVS     3$
10708  036742  103001                  BCC     4$
10709
10710  036744  104002          3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10711
10712  036746  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10713  036750  001401                  BEQ     TST546          ;;BR IF YES
10714
*10715  036752  104002          5$:     ERROR   2               ;CMP DELIVERED A RESULT
10716
10717                          ;;**********************************************************
10718                          ;*TEST 546      CMP TEST - SM1,DM0 - N:C = 0000
10719                          ;;**********************************************************
10720  036754                  TST546:
10721  036754  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10722  036756  012700  000546          MOV     #546,R0         ;;LOAD R0 WITH TEST NUMBER
10723  036762  013701  037010          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10724  036766  012704  000001          MOV     #+1,R4          ;RESULT S / B = +1
```

```
10725  036772  012705  063316          MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10726  036776  012703  000001          MOV     #+1,R3          ;[DEST] = +1
10727  037002  012715  100000          MOV     #100000,(R5)    ;SRC OPR = 100000
10728  037006  000257                  CCC                     ;CLEAR FLAGS
10729
10730  037010  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10731
10732  037012  100403                  BMI     3$              ;N:C = 0010
10733  037014  001402                  BEQ     3$
10734  037016  102001                  BVC     3$
10735  037020  103001                  BCC     4$
10736
10737  037022  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10738
10739  037024  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10740  037026  001401                  BEQ     TST547          ;;BR IF YES
10741
10742  037030  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10743
10744                                  ;;*****************************************************************
10745                                  ;*TEST 547      BIS  SM1,DM1 TEST - N:C = 0111
10746                                  ;;*****************************************************************
10747  037032                          TST547:
10748  037032  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10749  037034  012700  000547                  MOV     #547,R0         ;;LOAD R0 WITH TEST NUMBER
10750  037040  013701  037070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10751  037044  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10752  037050  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10753  037054  012705  063332                  MOV     #DWTA+10,R5     ;SOURCE ADDR = DWTA+10
10754  037060  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 052525
10755  037064  000257                          CCC                     ;CLEAR FLAGS
10756  037066  000267                          267                     ;N:C = 0111
10757
10758  037070  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10759
10760  037072  100003                          BPL     3$              ;N:C = 1001?
10761  037074  001402                          BEQ     3$
10762  037076  102401                          BVS     3$
10763  037100  103401                          BCS     4$
10764
10765  037102  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10766
10767  037104  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10768  037106  001402                          BEQ     TST550          ;;BR IF YES
10769
10770  037110  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10771  037112  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10772
10773                                  ;;*****************************************************************
10774                                  ;*TEST 550      BIS  SM1,DM1 TEST - N:C = 1000
10775                                  ;;*****************************************************************
10776  037114                          TST550:
10777  037114  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10778  037116  012700  000550                  MOV     #550,R0         ;;LOAD R0 WITH TEST NUMBER
10779  037122  013701  037146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10780  037126  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
10781  037132  005004                          CLR     R4              ;RESULT S / B = 000000
10782  037134  012705  063322                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
10783  037140  005012                          CLR     (R2)            ;[DEST] = 000000
10784  037142  000257                          CCC                     ;CLEAR FLAGS
10785  037144  000270                          SEN                     ;N:C = 1000
10786
10787  037146  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10788
10789  037150  100403                          BMI     3$              ;N:C = 0100 ?
10790  037152  001002                          BNE     3$
10791  037154  102401                          BVS     3$
10792  037156  103001                          BCC     4$
10793
10794  037160  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10795
10796  037162  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10797  037164  001402                          BEQ     TST551          ;;BR IF YES
10798
10799  037166  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10800  037170  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10801
10802                                  ;;**************************************************************
10803                                  ;*TEST 551      BIC  SM1,DM1 TEST - N:C = 0111
10804                                  ;;**************************************************************
10805  037172                          TST551:
10806  037172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10807  037174  012700  000551                  MOV     #551,R0         ;;LOAD R0 WITH TEST NUMBER
10808  037200  013701  037234                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10809  037204  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10810  037210  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10811  037214  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10812  037220  012715  077777                  MOV     #77777,(R5)     ;[SOURCE] = 77777
10813  037224  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
10814  037230  000257                          CCC                     ;CLEAR FLAGS
10815  037232  000267                          267                     ;N:C = 0111
10816
10817  037234  041512                  2$:     BIC     (R5),(R2)       ;TEST THE BIC
10818
10819  037236  100003                          BPL     3$              ;N:C = 1001 ?
10820  037240  001402                          BEQ     3$
10821  037242  102401                          BVS     3$
10822  037244  103401                          BCS     4$
10823
10824  037246  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10825
10826  037250  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10827  037252  001402                          BEQ     TST552          ;;BR IF YES
10828
10829  037254  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10830  037256  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10831
10832                                  ;;**************************************************************
10833                                  ;*TEST 552      BIC  SM1,DM1 TEST - N:C = 1000
10834                                  ;;**************************************************************
10835  037260                          TST552:
10836  037260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
10837  037262  012700  000552          MOV     #552,R0         ;;LOAD R0 WITH TEST NUMBER
10838  037266  013701  037314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10839  037272  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10840  037276  005004                  CLR     R4              ;RESULT S / B = 000000
10841  037300  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10842  037304  005015                  CLR     (R5)            ;[SOURCE] = 000000
10843  037306  005012                  CLR     (R2)            ;[DEST] = 000000
10844  037310  000257                  CCC                     ;CLEAR FLAGS
10845  037312  000270                  SEN                     ;N:C = 1000
10846
10847  037314  041512          2$:     BIC     (R5),(R2)       ;TEST THE BIC
10848
10849  037316  100403                  BMI     3$              ;N:C = 0100 ?
10850  037320  001002                  BNE     3$
10851  037322  102401                  BVS     3$
10852  037324  103001                  BCC     4$
10853
10854  037326  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10855
10856  037330  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10857  037332  001402                  BEQ     TST553          ;;BR IF YES
10858
10859  037334  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10860  037336  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10861
10862                          ;;**************************************************************
10863                          ;*TEST 553      BIT  SM1,DM1 TEST - N:C = 1000
10864                          ;;**************************************************************
10865  037340                  TST553:
10866  037340  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10867  037342  012700  000553          MOV     #553,R0         ;;LOAD R0 WITH TEST NUMBER
10868  037346  013701  037402          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10869  037352  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10870  037356  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
10871  037362  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10872  037366  012715  052525          MOV     #52525,(R5)     ;[SOURCE] = 052525
10873  037372  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
10874  037376  000257                  CCC                     ;CLEAR FLAGS
10875  037400  000270                  SEN                     ;N:C = 1000
10876
10877  037402  031512          2$:     BIT     (R5),(R2)       ;TEST THE BIT
10878
10879  037404  100403                  BMI     3$              ;N:C = 0100 ?
10880  037406  001002                  BNE     3$
10881  037410  102401                  BVS     3$
10882  037412  103001                  BCC     4$
10883
10884  037414  104001          3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10885
10886  037416  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10887  037420  001402                  BEQ     TST554          ;;BR IF YES
10888
10889  037422  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10890  037424  104001          5$:     ERROR   1               ;BIT DELIVERED A RESULT
10891
10892                          ;;**************************************************************
```

```
10893                                   ;*TEST 554      BIT  SM1,DM1 TEST - N:C = 0111
10894                                   ;;*****************************************************************
10895  037426                           TST554:
10896  037426  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10897  037430  012700  000554                   MOV     #554,R0         ;;LOAD R0 WITH TEST NUMBER
10898  037434  013701  037502                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10899                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT11
10900  037440  032737  004000  063234           BIT     #BIT11,@#BPTLOC ;BREAKPOINT HALT SET ??
10901  037446  001401                           BEQ     .+4             ;BR IF NOT
10902  037450  000000                           HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10903  037452  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10904  037456  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
10905  037462  012705  063316                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10906  037466  012715  100000                   MOV     #100000,(R5)    ;[SOURCE] = 100000
10907  037472  012712  100000                   MOV     #100000,(R2)    ;[DEST] = 100000
10908  037476  000257                           CCC                     ;CLEAR FLAGS
10909  037500  000267                           267                     ;N:C = 0111
10910
10911  037502  031512                   2$:     BIT     (R5),(R2)       ;TEST THE BIT
10912
10913  037504  100003                           BPL     3$              ;N:C = 1001 ?
10914  037506  001402                           BEQ     3$
10915  037510  102401                           BVS     3$
10916  037512  103401                           BCS     4$
10917
10918  037514  104001                   3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10919
10920  037516  020412                   4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10921  037520  001402                           BEQ     TST555          ;;BR IF YES
10922
10923  037522  011203                           MOV     (R2),R3         ;GET THE WAS DATA
10924  037524  104001                   5$:     ERROR   1               ;BIT DELIVERED A RESULT
10925
10926                                   ;;*****************************************************************
10927                                   ;*TEST 555      CMP  SM1,DM1 TEST - N:C = 1010
10928                                   ;;*****************************************************************
10929  037526                           TST555:
10930  037526  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10931  037530  012700  000555                   MOV     #555,R0         ;;LOAD R0 WITH TEST NUMBER
10932  037534  013701  037566                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10933  037540  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10934  037544  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
10935  037550  012705  063316                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10936  037554  012715  177777                   MOV     #-1,(R5)        ;[SOURCE] = 177777
10937  037560  010412                           MOV     R4,(R2)         ;[DEST] = 177777
10938  037562  000257                           CCC                     ;CLEAR FLAGS
10939  037564  000272                           272                     ;N:C = 1010
10940
10941  037566  021512                   2$:     CMP     (R5),(R2)       ;TEST THE CMP
10942
10943  037570  100403                           BMI     3$              ;N:C = 0100 ?
10944  037572  001002                           BNE     3$
10945  037574  102401                           BVS     3$
10946  037576  103001                           BCC     4$
10947
10948  037600  104001                   3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
```

```
10949
10950  037602  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10951  037604  001402                          BEQ     TST556          ;;BR IF YES
10952
10953  037606  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10954  037610  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10955
10956                                  ;;**************************************************************
10957                                  ;*TEST 556       CMP  SM1,DM1 TEST - N:C = 0110
10958                                  ;;**************************************************************
10959  037612                          TST556:
10960  037612  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10961  037614  012700  000556                  MOV     #556,R0         ;;LOAD R0 WITH TEST NUMBER
10962  037620  013701  037652                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10963  037624  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10964  037630  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
10965  037634  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10966  037640  005015                          CLR     (R5)            ;[SOURCE] = 000000
10967  037642  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
10968  037646  000257                          CCC                     ;CLEAR FLAGS
10969  037650  000266                          266                     ;N:C = 0110
10970
10971  037652  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
10972
10973  037654  100003                          BPL     3$              ;N:C = 1001 ?
10974  037656  001402                          BEQ     3$
10975  037660  102401                          BVS     3$
10976  037662  103401                          BCS     4$
10977
10978  037664  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10979
10980  037666  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10981  037670  001402                          BEQ     TST557          ;;BR IF YES
10982
10983  037672  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10984  037674  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10985
10986                                  ;;**************************************************************
10987                                  ;*TEST 557       CMP  SM1,DM1 TEST - N:C = 0000
10988                                  ;;**************************************************************
10989  037676                          TST557:
10990  037676  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10991  037700  012700  000557                  MOV     #557,R0         ;;LOAD R0 WITH TEST NUMBER
10992  037704  013701  037736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10993  037710  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10994  037714  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
10995  037720  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10996  037724  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 000000
10997  037730  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
10998  037734  000257                          CCC                     ;CLEAR FLAGS
10999
11000  037736  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
11001
11002  037740  100403                          BMI     3$              ;N:C = 0010 ?
11003  037742  001402                          BEQ     3$
11004  037744  102001                          BVC     3$
```

```
11005  037746  103001                          BCC     4$
11006
11007  037750  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
11008
11009  037752  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
11010  037754  001402                          BEQ     TST560          ;;BR IF YES
11011
11012  037756  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11013  037760  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
11014
11015                                  ;;**************************************************************
11016                                  ;*TEST 560      BISB  SM1,DM0 TEST - SOURCE ADDR ODD
11017                                  ;;**************************************************************
11018  037762                          TST560:
11019  037762  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11020  037764  012700  000560                  MOV     #560,R0         ;;LOAD R0 WITH TEST NUMBER
11021  037770  013701  040010                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11022  037774  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11023  040000  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11024  040004  005003                          CLR     R3              ;[DEST] = 000000
11025  040006  000257                          CCC                     ;SCOPE SYNC
11026
11027  040010  151503                  2$:     BISB    (R5),R3         ;TEST THE BISB
11028
11029  040012  020403                          CMP     R4,R3           ;RESULT CORRECT ?
11030  040014  001401                          BEQ     TST561          ;;BR IF YES
11031
11032  040016  104002                  3$:     ERROR   2               ;BISB DELIVERED THE WRONG RESULT
11033
11034                                  ;;**************************************************************
11035                                  ;*TEST 561      BISB  SM1,DM1 TEST - SOURCE ADDR ODD
11036                                  ;;**************************************************************
11037  040020                          TST561:
11038  040020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11039  040022  012700  000561                  MOV     #561,R0         ;;LOAD R0 WITH TEST NUMBER
11040  040026  013701  040052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11041  040032  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11042  040036  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11043  040042  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11044  040046  005012                          CLR     (R2)            ;[DEST] = 000000
11045  040050  000257                          CCC                     ;SCOPE SYNC
11046
11047  040052  151512                  2$:     BISB    (R5),(R2)       ;TEST THE BISB
11048
11049  040054  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11050  040056  001402                          BEQ     TST562          ;;BR IF YES
11051
11052  040060  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11053  040062  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11054
11055                                  ;;**************************************************************
11056                                  ;*TEST 562      BISB  SM1,DM2 TEST - SOURCE ADDR ODD
11057                                  ;;**************************************************************
11058  040064                          TST562:
11059  040064  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11060  040066  012700  000562                  MOV     #562,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
11061  040072  013701  040120                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11062  040076  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11063  040102  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11064  040106  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11065  040112  005012                          CLR     (R2)            ;[DEST] = 000000
11066  040114  010203                          MOV     R2,R3           ;DEST ADDR IN R3
11067  040116  000257                          CCC                     ;SCOPE SYNC
11068
11069  040120  151523                  2$:     BISB    (R5),(R3)+      ;TEST THE BISB
11070
11071  040122  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11072  040124  001402                          BEQ     TST563          ;;BR IF YES
11073
11074  040126  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11075  040130  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11076
11077                                   ;;*****************************************************************
11078                                   ;*TEST 563      BISB  SM1,DM3 TEST - SOURCE ADDR ODD
11079                                   ;;*****************************************************************
11080  040132                          TST563:
11081  040132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11082  040134  012700  000563                  MOV     #563,R0         ;;LOAD R0 WITH TEST NUMBER
11083  040140  013701  040170                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11084  040144  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11085  040150  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11086  040154  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11087  040160  005012                          CLR     (R2)            ;[DEST] = 000000
11088  040162  012703  063306                  MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
11089  040166  000257                          CCC                     ;SCOPE SYNC
11090
11091  040170  151533                  2$:     BISB    (R5),@(R3)+     ;TEST THE BISB
11092
11093  040172  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11094  040174  001402                          BEQ     TST564          ;;BR IF YES
11095
11096  040176  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11097  040200  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11098
11099                                   ;;*****************************************************************
11100                                   ;*TEST 564      BISB  SM1,DM4 TEST - SOURCE ADDR ODD
11101                                   ;;*****************************************************************
11102  040202                          TST564:
11103  040202  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11104  040204  012700  000564                  MOV     #564,R0         ;;LOAD R0 WITH TEST NUMBER
11105  040210  013701  040240                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11106  040214  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11107  040220  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
11108  040224  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11109  040230  012703  063314                  MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
11110  040234  005012                          CLR     (R2)            ;[DEST] = 000000
11111  040236  000257                          CCC                     ;SCOPE SYNC
11112
11113  040240  151543                  2$:     BISB    (R5),-(R3)      ;TEST THE BISB
11114
11115  040242  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11116  040244  001402                          BEQ     TST565          ;;BR IF YES
```

```
11117
11118  040246  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11119  040250  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11120
11121                                    ;;*****************************************************************
11122                                    ;*TEST 565      BISB  SM1,DM5 TEST - SOURCE ADDR ODD
11123                                    ;;*****************************************************************
11124  040252                            TST565:
11125  040252  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11126  040254  012700  000565            MOV     #565,R0         ;;LOAD R0 WITH TEST NUMBER
11127  040260  013701  040310            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11128  040264  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11129  040270  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
11130  040274  012705  064631            MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11131  040300  012703  063310            MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
11132  040304  005012                    CLR     (R2)            ;[DEST] = 000000
11133  040306  000257                    CCC                     ;SCOPE SYNC
11134
11135  040310  151553            2$:     BISB    (R5),@-(R3)     ;TEST THE BISB
11136
11137  040312  020412                    CMP     R4,(R2)         ;CORRECT RESULT
11138  040314  001402                    BEQ     TST566          ;;BR IF YES
11139
11140  040316  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11141  040320  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11142
11143                                    ;;*****************************************************************
11144                                    ;*TEST 566      BISB  SM1,DM6 TEST - SOURCE ADDR ODD
11145                                    ;;*****************************************************************
11146  040322                            TST566:
11147  040322  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11148  040324  012700  000566            MOV     #566,R0         ;;LOAD R0 WITH TEST NUMBER
11149  040330  013701  040360            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11150  040334  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11151  040340  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
11152  040344  012705  064631            MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11153  040350  012703  063320            MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
11154  040354  005012                    CLR     (R2)            ;[DEST] = 000000
11155  040356  000257                    CCC                     ;SCOPE SYNC
11156
11157  040360  151563  177772    2$:     BISB    (R5),-6(R3)     ;TEST THE BISB
11158
11159  040364  020412                    CMP     R4,(R2)         ;CORRECT RESULT
11160  040366  001402                    BEQ     TST567          ;;BR IF YES
11161
11162  040370  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11163  040372  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11164
11165                                    ;;*****************************************************************
11166                                    ;*TEST 567      BISB  SM1,DM7 TEST - SOURCE ADDR ODD
11167                                    ;;*****************************************************************
11168  040374                            TST567:
11169  040374  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11170  040376  012700  000567            MOV     #567,R0         ;;LOAD R0 WITH TEST NUMBER
11171  040402  013701  040432            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11172  040406  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
11173  040412  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
11174  040416  012705  064631                MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11175  040422  012703  063276                MOV     #ATA,R3         ;BASE DEST ADDR = ATA
11176  040426  005012                        CLR     (R2)            ;[DEST] = 000000
11177  040430  000257                        CCC                     ;SCOPE SYNC
11178
11179  040432  151573  000010        2$:     BISB    (R5),@10(R3)    ;TEST THE BISB
11180
11181  040436  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11182  040440  001402                        BEQ     TST570          ;;BR IF YES
11183
11184  040442  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11185  040444  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11186
11187                                ;;*****************************************************************
11188                                ;*TEST 570      BISB  SM0,DM2 TEST - DEST ADDR EVEN
11189                                ;;*****************************************************************
11190  040446                        TST570:
11191  040446  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11192  040450  012700  000570                MOV     #570,R0         ;;LOAD R0 WITH TEST NUMBER
11193  040454  013701  040476                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11194  040460  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11195  040464  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
11196  040470  010203                        MOV     R2,R3           ;DEST ADDR IN R3
11197  040472  005012                        CLR     (R2)            ;[DEST] = 000000
11198  040474  000257                        CCC                     ;SCOPE SYNC
11199
11200  040476  150423                2$:     BISB    R4,(R3)+        ;TEST THE BISB
11201
11202  040500  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11203  040502  001402                        BEQ     TST571          ;;BR IF YES
11204
11205  040504  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11206  040506  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11207
11208                                ;;*****************************************************************
11209                                ;*TEST 571      BISB  SM0,DM1 TEST - DEST ADDR ODD
11210                                ;;*****************************************************************
11211  040510                        TST571:
11212  040510  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11213  040512  012700  000571                MOV     #571,R0         ;;LOAD R0 WITH TEST NUMBER
11214  040516  013701  040546                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11215  040522  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11216  040526  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
11217  040532  012705  000377                MOV     #377,R5         ;[R5]=SOURCE OPR = 377
11218  040536  012703  063313                MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
11219  040542  005012                        CLR     (R2)            ;[DEST] = 000000
11220  040544  000257                        CCC                     ;SCOPE SYNC
11221
11222  040546  150513                2$:     BISB    R5,(R3)         ;TEST THE BISB
11223
11224  040550  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11225  040552  001402                        BEQ     TST572          ;;BR IF YES
11226
11227  040554  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11228  040556  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
```

```
11229
11230                                   ;;*****************************************************************
11231                                   ;*TEST 572      BISB  SM0,DM1 TEST - DEST ADDR EVEN
11232                                   ;;*****************************************************************
11233  040560                           TST572:
11234  040560  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11235  040562  012700  000572                  MOV     #572,R0         ;;LOAD R0 WITH TEST NUMBER
11236  040566  013701  040610                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11237  040572  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11238  040576  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
11239  040602  010203                          MOV     R2,R3           ;DEST ADDR IN R3
11240  040604  005012                          CLR     (R2)            ;[DEST] = 000000
11241  040606  000257                          CCC                     ;SCOPE SYNC
11242
11243  040610  150413                  2$:     BISB    R4,(R3)         ;TEST THE BISB
11244
11245  040612  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11246  040614  001402                          BEQ     TST573          ;;BR IF YES
11247
11248  040616  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11249  040620  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11250
11251                                   ;;*****************************************************************
11252                                   ;*TEST 573      BISB  SM1,DM1 TEST - DEST ADDR ODD
11253                                   ;;*****************************************************************
11254  040622                           TST573:
11255  040622  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11256  040624  012700  000573                  MOV     #573,R0         ;;LOAD R0 WITH TEST NUMBER
11257  040630  013701  040660                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11258  040634  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11259  040640  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
11260  040644  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11261  040650  012703  063313                  MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
11262  040654  005012                          CLR     (R2)            ;[DEST] = 000000
11263  040656  000257                          CCC                     ;SCOPE SYNC
11264
11265  040660  151513                  2$:     BISB    (R5),(R3)       ;TEST THE BISB
11266
11267  040662  020412                          CMP     R4,(R2)         ;CORRECT RESULT
11268  040664  001402                          BEQ     TST574          ;;BR IF YES
11269
11270  040666  011203                          MOV     (R2),R3         ;GET THE WAS DATA
11271  040670  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11272
11273                                   ;;*****************************************************************
11274                                   ;*TEST 574      JMP MODE 1 TEST, FLAGS = 1111
11275                                   ;;*****************************************************************
11276  040672                           TST574:
11277  040672  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11278  040674  012700  000574                  MOV     #574,R0         ;;LOAD R0 WITH TEST NUMBER
11279  040700  013701  040712                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11280  040704  012702  040720                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11281  040710  000277                          SCC                     ;MAKE N:C = 1111
11282
11283  040712  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11284
```

```
11285  040714  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11286  040716  000405                          BR      TST575          ;;GO CALL SCOPE
11287
11288  040720  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11289  040722  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11290  040724  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11291  040726  100401                          BMI     TST575          ;;BR IF "N" STILL SET
11292
11293  040730  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED ONE
11294
11295                                  ;;**********************************************************************
11296                                  ;*TEST 575      JMP MODE 1 TEST, FLAGS = 0000
11297                                  ;;**********************************************************************
11298  040732                          TST575:
11299  040732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11300  040734  012700  000575                  MOV     #575,R0         ;;LOAD R0 WITH TEST NUMBER
11301  040740  013701  040752                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11302  040744  012702  040760                  MOV     #4$,R2          ;R2 CONTAINS JMP ADDRESS
11303  040750  000257                          CCC                     ;MAKE N:C = 0000
11304
11305  040752  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11306
11307  040754  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11308  040756  000405                          BR      TST576          ;;GO CALL SCOPE
11309
11310  040760  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11311  040762  102402                          BVS     5$              ;BR IF JMP SET "V"
11312  040764  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11313  040766  100001                          BPL     TST576          ;;BR IF "N" STILL CLEAR
11314
11315  040770  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET ONE
11316
11317                                  ;;**********************************************************************
11318                                  ;*TEST 576      JMP MODE 2 TEST; FLAGS = 1111
11319                                  ;;**********************************************************************
11320  040772                          TST576:
11321  040772  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11322  040774  012700  000576                  MOV     #576,R0         ;;LOAD R0 WITH TEST NUMBER
11323  041000  013701  041012                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11324  041004  012702  041020                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11325  041010  000277                          SCC                     ;SET N:C = 1111
11326
11327  041012  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11328
11329  041014  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11330  041016  000411                          BR      TST577          ;;GO TO SCOPE EXIT
11331
11332  041020  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11333  041022  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11334  041024  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11335  041026  100401                          BMI     6$              ;BR IF "N" STILL SET
11336
11337  041030  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11338
11339  041032  022702  041022          6$:     CMP     #4$+2,R2        ;DID R2 GET AUTO-INCREMENTED?
11340  041036  001401                          BEQ     TST577          ;;BR IF YES
```

```
11341
11342  041040  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER (R2)
11343
11344                                  ;;*****************************************************************
11345                                  ;*TEST 577      JMP MODE 2 TEST; FLAGS = 0000
11346                                  ;;*****************************************************************
11347  041042                          TST577:
11348  041042  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11349  041044  012700  000577                  MOV     #577,R0         ;;LOAD R0 WITH TEST NUMBER
11350  041050  013701  041062                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11351  041054  012702  041070                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11352  041060  000257                          CCC                     ;MAKE N:C = 0000
11353
11354  041062  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11355
11356  041064  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11357  041066  000405                          BR      TST600          ;;GO TO SCOPE EXIT
11358
11359  041070  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11360  041072  102402                          BVS     5$              ;BR IF JMP SET "V"
11361  041074  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11362  041076  100001                          BPL     TST600          ;;BR IF "N" IS CLEAR
11363
11364  041100  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11365
11366                                  ;;*****************************************************************
11367                                  ;*TEST 600      JMP TEST MODE 3; FLAGS = 1111
11368                                  ;;*****************************************************************
11369  041102                          TST600:
11370  041102  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11371  041104  012700  000600                  MOV     #600,R0         ;;LOAD R0 WITH TEST NUMBER
11372  041110  013701  041122                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11373  041114  012702  041154                  MOV     #7$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11374  041120  000277                          SCC                     ;SET N:C = 1111
11375
11376  041122  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11377
11378  041124  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11379  041126  000414                          BR      TST601          ;;GO TO SCOPE EXIT
11380
11381  041130  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11382  041132  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11383  041134  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11384  041136  100401                          BMI     6$              ;BR IF "N" STILL SET
11385
11386  041140  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEAR
11387
11388  041142  022702  041156          6$:     CMP     #7$+2,R2        ;DID JMP UPDATE R2?
11389  041146  001404                          BEQ     TST601          ;;BR IF YES
11390
11391  041150  104006                          ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11392  041152  000402                          BR      TST601          ;;GO TO SCOPE EXIT
11393  041154  041130                  7$:     4$                      ;JMP3 CONTAINS JUMP ADDRESS
11394  041156  104006                          ERROR   6               ;ERROR CALL OCCURS IF MODE3 HAPPENS
11395                                                                  ;TO EXECUTE AS MODE 1 OR 2 AND
11396                                                                  ;4$ IS LEGAL INSTRUCTION
```

```
11397
11398                                   ;;*****************************************************************
11399                                   ;*TEST 601      JMP TEST MODE 3; FLAGS = 0000
11400                                   ;;*****************************************************************
11401  041160                           TST601:
11402  041160  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11403  041162  012700  000601                  MOV     #601,R0         ;;LOAD R0 WITH TEST NUMBER
11404  041166  013701  041200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11405  041172  012702  041222                  MOV     #6$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11406  041176  000257                          CCC                     ;MAKE N:C = 0000
11407
11408  041200  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11409
11410  041202  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
11411  041204  000410                          BR      TST602          ;;GO TO SCOPE EXIT
11412
11413  041206  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11414  041210  102402                          BVS     5$              ;BR IF JMP SET "V"
11415  041212  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11416  041214  100004                          BPL     TST602          ;;BR IF "N" STILL CLEAR
11417
11418  041216  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11419  041220  000402                          BR      TST602          ;;GO TO SCOPE EXIT
11420
11421  041222  041206                  6$:     4$                      ;JUMP ADDRESS IN 6$
11422  041224  104006                          ERROR   6               ;JMP MODE 3 EXECUTED LIKE MODE 1 OR 2
11423
11424                                   ;;*****************************************************************
11425                                   ;*TEST 602      JMP TEST MODE 4; FLAGS = 1111
11426                                   ;;*****************************************************************
11427  041226                           TST602:
11428  041226  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11429  041230  012700  000602                  MOV     #602,R0         ;;LOAD R0 WITH TEST NUMBER
11430  041234  013701  041246                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11431  041240  012702  041256                  MOV     #5$,R2  ;(R2) = JMP ADDRESS PLUS 2
11432  041244  000277                          SCC                     ;MAKE N:C = 1111
11433
11434  041246  000142                  2$:     JMP     -(R2)           ;TEST THE JMP - GO TO 5$ MINUS 2
11435
11436  041250  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11437  041252  000414                          BR      TST603          ;;GO TO SCOPE EXIT
11438
11439  041254  000402                          BR      4$              ;GO TEST FLAGS - JMP LOADED PC OK
11440  041256  104006                  5$:     ERROR   6               ;JMP FAILED TO AUTO-DECREMENT R2
11441  041260  000411                          BR      TST603          ;;GO TO SCOPE EXIT
11442
11443  041262  103003                  4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11444  041264  102002                          BVC     7$              ;BR IF JMP CLEARED "V"
11445  041266  001001                          BNE     7$              ;BR IF JMP CLEARED "Z"
11446  041270  100401                          BMI     6$              ;BR IF "N" STILL SET
11447
11448  041272  104006                  7$:     ERROR   6               ;JMP ALTERED FLAGS
11449
11450  041274  022702  041254          6$:     CMP     #5$-2,R2        ;DID JMP UPDATE R2 PROPERLY?
11451  041300  001401                          BEQ     TST603          ;;BR IF YES
11452
```

```
11453  041302  104006                  9$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11454
11455                                  ;;*****************************************************************
11456                                  ;*TEST 603      JMP TEST MODE 4; FLAGS = 0000
11457                                  ;;*****************************************************************
11458  041304                          TST603:
11459  041304  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11460  041306  012700  000603                 MOV     #603,R0         ;;LOAD R0 WITH TEST NUMBER
11461  041312  013701  041324                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11462  041316  012702  041334                 MOV     #4$+2,R2        ;(R2) = JUMP ADDRESS PLUS 2
11463  041322  000257                         CCC                     ;MAKE N:C = 0000
11464
11465  041324  000142                  2$:     JMP     -(R2)           ;TEST THE JMP - TO TO 4$
11466
11467  041326  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11470  041330  000405                         BR      TST604          ;;GO TO SCOPE EXIT
11469
11470  041332  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11471  041334  102402                          BVS     5$              ;BR IF JMP SET "V"
11472  041336  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11473  041340  100001                          BPL     TST604          ;;BR IF "N" STILL CLEAR
11474
11475  041342  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11476
11477                                  ;;*****************************************************************
11478                                  ;*TEST 604      JMP TEST MODE 5; FLAGS = 1111
11479                                  ;;*****************************************************************
11480  041344                          TST604:
11481  041344  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11482  041346  012700  000604                 MOV     #604,R0         ;;LOAD R0 WITH TEST NUMBER
11483  041352  013701  041364                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11484  041356  012702  041420                 MOV     #JMP$,R2        ;JMP CONTAINS ADDR+2 OF JUMP ADDRESS
11485  041362  000277                         SCC
11486
11487  041364  000152                  2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11488
11489  041366  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11490  041370  000414                         BR      TST605          ;;GO TO SCOPE OXIT
11491
11492  041372  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11493  041374  102002                          BVC     5$
11494  041376  001001                          BNE     5$
11495  041400  100401                          BMI     6$
11496
11497  041402  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11498
11499  041404  022702  041416          6$:     CMP     #JMP$-2,R2      ;DID R2 GET AUTO-DECREMENTED
11500  041410  001404                          BEQ     TST605          ;;BR IF YES
11501
11502  041412  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11503  041414  000402                         BR      TST605          ;;GO TO SCOPE EXIT
11504  041416  041372                          4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11505  041420  104006                  JMP$:   ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11506
11507                                  ;;*****************************************************************
11508                                  ;*TEST 605      JMP TEST MODE 5; FLAG = 0000
```

```
11509                                   ;;*****************************************************************
11510  041422                           TST605:
11511  041422  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11512  041424  012700  000605                   MOV     #605,R0         ;;LOAD R0 WITH TEST NUMBER
11513  041430  013701  041442                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11514  041434  012702  041466                   MOV     #JMPSA,R2       ;(R2) = ADDR +2 OF JUMP ADDRESS
11515  041440  000257                           CCC                     ;SET N:C = 0000
11516
11517  041442  000152                   2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11518
11519  041444  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11520  041446  000410                           BR      TST606          ;;GO TO SCOPE EXIT
11521
11522  041450  103403                   4$:     BCS     5$              ;BR IF JMP SET "C"
11523  041452  102402                           BVS     5$              ;BR IF JMP SET "V"
11524  041454  001401                           BEQ     5$              ;BR IF JMP SET "Z"
11525  041456  100004                           BPL     TST606          ;;BR IF "N" STILL CLEAR
11526
11527  041460  104006                   5$:     ERROR   6               ;JMP ALTERED THE CODES - SET
11528  041462  000402                           BR      TST606          ;;GO TO SCOPE EXIT
11529
11530  041464  041450                           4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11531  041466  104006                   JMPSA:  ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11532
11533                                   ;;*****************************************************************
11534                                   ;;TEST 606      JMP TEST MODE 6; FLAGS = 1111
11535                                   ;;*****************************************************************
11536  041470                           TST606:
11537  041470  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11538  041472  012700  000606                   MOV     #606,R0         ;;LOAD R0 WITH TEST NUMBER
11539  041476  013701  041510                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11540  041502  012702  041534                   MOV     #7$,R2  ;(R2) = BASE ADDRESS TO BE INDEXED
11541  041506  000277                           SCC                     ;MAKE N:C = 1111
11542
11543  041510  000162  177764           2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11544
11545  041514  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
11546  041516  000407                           BR      TST607          ;;GO TO SCOPE EXIT
11547
11548  041520  103003                   4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11549  041522  102002                           BVC     5$
11550  041524  001001                           BNE     5$
11551  041526  100403                           BMI     TST607          ;;BR IF "N" STILL SET
11552
11553  041530  104006                   5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11554  041532  000401                           BR      TST607          ;;GO TO SCOPE EXIT
11555
11556  041534  104006                   7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2 OR
11557                                                                   ;FAILED TO INDEX (R2)
11558
11559                                   ;;*****************************************************************
11560                                   ;;TEST 607      JMP TEST MODE 6; FLAGS = 0000
11561                                   ;;*****************************************************************
11562  041536                           TST607:
11563  041536  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11564  041540  012700  000607                   MOV     #607,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
11565  041544  013701  041556                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11566  041550  012702  041602                  MOV     #7$,R2  ;[R2] = BASE ADDRESS FOR JUMP
11567  041554  000257                          CCC                     ;MAKE N:C = 0000
11568
11569  041556  000162  177764          2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11570
11571  041562  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11572  041564  000407                          BR      TST610          ;;GO TO SCOPE EXIT
11573
11574  041566  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11575  041570  102402                          BVS     5$              ;BR IF JMP SET "V"
11576  041572  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11577  041574  100003                          BPL     TST610          ;;BR IF "N" STILL CLEAR
11578
11579  041576  104006                  5$:     ERROR   6               ;JMP ALTERED CODES
11580  041600  000401                          BR      TST610          ;;GO TO SCOPE EXIT
11581
11582  041602  104006                  7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2, OR
11583                                                                  ;FAILED TO INDEX [R2]
11584
11585                                  ;;**************************************************************
11586                                  ;*TEST 610      JMP TEST MODE 7; FLAGS = 1111
11587                                  ;;**************************************************************
11588  041604                          TST610:
11589  041604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11590  041606  012700  000610                  MOV     #610,R0         ;;LOAD R0 WITH TEST NUMBER
11591  041612  013701  041624                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11592  041616  012702  041634                  MOV     #5$,R2  ;[R2] = BASE ADDRESS
11593  041622  000277                          SCC                     ;MAKE N:C = 1111
11594
11595  041624  000172  000020          2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11596
11597  041630  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11598  041632  000412                          BR      TST611          ;;GO TO SCOPE EXIT
11599
11600  041634  104006                  5$:     ERROR   6               ;JMP FAILED TO INDEX OR ACTED LIKE MODE 1 OR 2
11601  041636  000410                          BR      TST611          ;;GO TO SCOPE EXIT
11602
11603  041640  103003                  4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11604  041642  102002                          BVC     7$              ;BR IF JMP CLEARED "V"
11605  041644  001001                          BNE     7$              ;BR IF JMP CLEARED "Z"
11606  041646  100404                          BMI     TST611          ;;BR IF "N" STILL SET
11607
11608  041650  104006                  7$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11609  041652  000402                          BR      TST611          ;;GO TO SCOPE EXIT
11610
11611  041654  041640                  8$:     4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11612
11613  041656  104006                          ERROR   6               ;JMP EXECUTED LIKE MODE 6
11614
11615                                  ;;**************************************************************
11616                                  ;*TEST 611      JMP TEST MODE 7; FLAGS = 0000
11617                                  ;;**************************************************************
11618  041660                          TST611:
11619  041660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11620  041662  012700  000611                  MOV     #611,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
11621  041666  013701  041700          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11622  041672  012702  041710          MOV     #5$,R2  ;(R2) = BASE ADDRESS
11623  041676  000257                  CCC                     ;MAKE N:C = 0000
11624
11625  041700  000172  000020  2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11626
11627  041704  104006          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11628  041706  000412                  BR      TST612          ;;GO TO SCOPE EXIT
11629
11630  041710  104006          5$:     ERROR   6               ;JMP FAILED TO INDEX
11631  041712  000410                  BR      TST612          ;;GO TO SCOPE EXIT
11632
11633  041714  103403          4$:     BCS     7$              ;BR IF JMP SET "C"
11634  041716  102402                  BVS     7$              ;BR IF JMP SET "V"
11635  041720  001401                  BEQ     7$              ;BR IF JMP SET "Z"
11636  041722  100004                  BPL     TST612          ;;BR IF "N" STILL CLEAR
11637
11638  041724  104006          7$:     ERROR   6               ;JMP ALTERED CODES - SET
11639  041726  000402                  BR      TST612          ;;GO TO SCOPE EXIT
11640
11641  041730  041714          8$:     4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11642
11643  041732  104006                  ERROR   6               ;JMP EXECUTED LIKE A MODE 6
11644
11645                           ;;**************************************************************
11646                           ;*TEST 612      JSR MODE 1 TEST - LOAD PC  /  PUSH SP
11647                           ;;**************************************************************
11648  041734                  TST612:
11649  041734  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11650  041736  012700  000612          MOV     #612,R0         ;;LOAD R0 WITH TEST NUMBER
11651  041742  013701  041764          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11652  041746  010605                  MOV     SP,R5           ;SAVE THE SP
11653  041750  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11654  041754  010506          1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11655  041756  012702  041770          MOV     #4$,R2          ;DEST ADDR = 4$
11656  041762  000257                  CCC                     ;SCOPE SYNC
11657
11658  041764  004412          2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11659
11660  041766  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11661
11662  041770  005726          4$:     TST     (SP)+           ;POP THE SP
11663  041772  020605                  CMP     SP,R5           ;DID JSR PUSH THE SP ?
11664  041774  001406                  BEQ     TST613          ;;BR IF YES
11665
11666  041776  005746                  TST     -(SP)           ;RESTORE ERROR SP
11667  042000  010603                  MOV     SP,R3           ;[R3]= WAS SP
11668  042002  010504                  MOV     R5,R4
11669  042004  005744                  TST     -(R4)           ;[R4]= S/B SP
11670  042006  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH THE SP
11671
11672  042010  010506                  MOV     R5,SP           ;RESTORE SP IN CASE OF ERROR
11673                           ;;**************************************************************
11674                           ;*TEST 613      JSR MODE 1 TEST - CHECK RN AND OLD PC
11675                           ;;**************************************************************
11676  042012                  TST613:
```

```
11677  042012  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11678  042014  012700  000613                MOV     #613,R0         ;;LOAD R0 WITH TEST NUMBER
11679  042020  013701  042052                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11680  042024  010605                        MOV     SP,R5           ;SAVE THE SP
11681  042026  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11682  042032  010506          1$:           MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11683  042034  012702  042056                MOV     #4$,R2          ;DEST ADDR = 4$
11684  042040  005066  177776                CLR     -2(SP)          ;INIT STACK LOC TO GET [R4]
11685  042044  012704  125252                MOV     #125252,R4      ;INIT RN = 125252
11686  042050  000257                        CCC                     ;SCOPE SYNC
11687
11688  042052  004412          2$:           JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11689
11690  042054  104006          3$:           ERROR   6               ;JSR FAILED TO LOAD THE PC
11691
11692  042056  022726  125252  4$:           CMP     #125252,(SP)+   ;DID JSR SAVE REG ON STACK
11693  042062  001401                        BEQ     8$              ;BR IF IT DID
11694
11695  042064  104005          5$:           ERROR   5               ;JSR FAILED TO SAVE REG ON STACK
11696
11697  042066  022704  042054  8$:           CMP     #3$,R4          ;DID OLD PC GET SAVED ?
11698  042072  001401                        BEQ     6$              ;BR IF YES
11699
11700  042074  104005          7$:           ERROR   5               ;JSR FAILED TO SAVE TH OLD PC
11701
11702  042076  010506          6$:           MOV     R5,SP           ;RESTORE SP IN CASE ERROR SCREWED IT UP
11703
11704                                        ;;*********************************************************************
11705                                        ;*TEST 614      JSR MODE 1 TEST - N:C = 0000
11706                                        ;;*********************************************************************
11707  042100                                TST614:
11708  042100  000004                               SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11709  042102  012700  000614                       MOV     #614,R0         ;;LOAD R0 WITH TEST NUMBER
11710  042106  013701  042142                       MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11711                                        .SBTTL USER CONTROLLED BREAKPOINT -- BIT12
11712  042112  032737  010000  063234               BIT     #BIT12,@#BPTLOC ;BREAKPOINT HALT SET ??
11713  042120  001401                               BEQ     .+4             ;BR IF NOT
11714  042122  000000                               HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
11715  042124  010605                               MOV     SP,R5           ;SAVE THE SP
11716  042126  010737  001010                       MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11717  042132  010506          1$:                  MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11718  042134  012702  042146                       MOV     #4$,R2          ;DEST ADDR = 4$
11719  042140  000257                               CCC                     ;N:C = 0000
11720
11721  042142  004412          2$:                  JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11722
11723  042144  104006          3$:                  ERROR   6               ;JSR FAILED TO LOAD THE PC
11724
11725  042146  100403          4$:                  BMI     5$              ;N:C = 0000 ?
11726  042150  001402                               BEQ     5$
11727  042152  102401                               BVS     5$
11728  042154  103001                               BCC     6$
11729
11730  042156  104005          5$:                  ERROR   5               ;JSR FAILED - ALTERED FLAGS
11731
11732  042160  010506          6$:                  MOV     R5,SP           ;RESET SP IN CASE OF ERROR
```

```
11733                                   ;;*****************************************************************
11734                                   ;*TEST 615      JSR MODE 1 TEST - N:C = 1111
11735                                   ;;*****************************************************************
11736  042162                           TST615:
11737  042162  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11738  042164  012700  000615                  MOV     #615,R0         ;;LOAD R0 WITH TEST NUMBER
11739  042170  013701  042212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11740  042174  010605                          MOV     SP,R5           ;SAVE THE SP
11741  042176  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
11742  042202  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11743  042204  012702  042216                  MOV     #4$,R2          ;DEST ADDR = 4$
11744  042210  000277                          SCC                     ;N:C = 1111
11745
11746  042212  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11747
11748  042214  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11749
11750  042216  100003                  4$:     BPL     5$              ;N:C = 1111 ?
11751  042220  001002                          BNE     5$
11752  042222  102001                          BVC     5$
11753  042224  103401                          BCS     6$
11754  042226  104005                  5$:     ERROR   5               ;JSR ALTERED FLAGS
11755
11756  042230  010506                  6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11757
11758                                   ;;*****************************************************************
11759                                   ;*TEST 616      JSR MODE 2 TEST
11760                                   ;;*****************************************************************
11761  042232                           TST616:
11762  042232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11763  042234  012700  000616                  MOV     #616,R0         ;;LOAD R0 WITH TEST NUMBER
11764  042240  013701  042262                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11765  042244  010605                          MOV     SP,R5           ;SAVE THE SP
11766  042246  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
11767  042252  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11768  042254  012702  042266                  MOV     #4$,R2          ;DEST ADDR = 4$
11769  042260  000257                          CCC                     ;SCOPE SYNC
11770
11771  042262  004422                  2$:     JSR     R4,(R2)+        ;TEST THE JSR - GO TO 4$
11772
11773  042264  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11774
11775  042266  005726                  4$:     TST     (SP)+           ;RESET SP
11776  042270  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11777  042272  001406                          BEQ     TST617          ;;BR IF YES
11778
11779  042274  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11780  042276  010603                          MOV     SP,R3           ;WAS SP
11781  042300  010504                          MOV     R5,R4
11782  042302  005744                          TST     -(R4)           ;S/B SP
11783  042304  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11784
11785  042306  010506                          MOV     R5,SP           ;RESTORE SP JUST IN CASE
11786
11787                                   ;;*****************************************************************
11788                                   ;*TEST 617      JSR MODE 3 TEST
```

```
11789                                   ;;*****************************************************************
11790  042310                           ;TST617:
11791  042310  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11792  042312  012700  000617                   MOV     #617,R0         ;;LOAD R0 WITH TEST NUMBER
11793  042316  013701  042340                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11794  042322  010605                           MOV     SP,R5           ;SAVE THE SP
11795  042324  010737  001010                   MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
11796  042330  010506                   1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11797  042332  012702  042366                   MOV     #7$,R2          ;DEST ADDR = [7$]
11798  042336  000257                           CCC                     ;SCOPE SYNC
11799
11800  042340  004432                   2$:     JSR     R4,@(R2)+       ;TEST THE JSR - GO TO 4$ VIA 7$
11801
11802  042342  104006                   3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11803
11804  042344  005726                   4$:     TST     (SP)+           ;RESET SP
11805  042346  020605                           CMP     SP,R5           ;DID JSR PUSH STACK ?
11806  042350  001411                           BEQ     TST620          ;;BR IF YES
11807
11808  042352  005746                           TST     -(SP)           ;RESET SP TO ERROR VALUE
11809  042354  010603                           MOV     SP,R3           ;WAS SP
11810  042356  010504                           MOV     R5,R4
11811  042360  005744                           TST     -(R4)           ;S/B SP
11812  042362  104003                   5$:     ERROR   3               ;JSR FAILED
11813  042364  000402                           BR      6$              ;GO EXIT
11814
11815  042366  042344                   7$:     4$                      ;CONTAINS JUMP ADDR
11816  042370  104006                           ERROR   6               ;JSR EXECUTED LIKE A MODE 1 OR 2
11817
11818  042372  010506                   6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11819
11820                                   ;;*****************************************************************
11821                                   ;*TEST 620      JSR MODE 4 TEST
11822                                   ;;*****************************************************************
11823  042374                           TST620:
11824  042374  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11825  042376  012700  000620                   MOV     #620,R0         ;;LOAD R0 WITH TEST NUMBER
11826  042402  013701  042424                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11827  042406  010605                           MOV     SP,R5           ;SAVE THE SP
11828  042410  010737  001010                   MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
11829  042414  010506                   1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11830  042416  012702  042432                   MOV     #5$,R2          ;DEST ADDR = 4$+2
11831  042422  000257                           CCC                     ;SCOPE SYNC
11832
11833  042424  004442                   2$:     JSR     R4,-(R2)        ;TEST THE JSR - GO TO 4$
11834
11835  042426  104006                   3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11836
11837  042430  000401                   4$:     BR      6$              ;JUMPED OK - GO CHECK SP
11838  042432  104005                   5$:     ERROR   5               ;JSR FAILED TO DECREMENT DEST REG
11839
11840  042434  005726                   6$:     TST     (SP)+           ;RESET SP
11841  042436  020605                           CMP     SP,R5           ;DID JSR PUSH STACK ?
11842  042440  001406                           BEQ     TST621          ;;BR IF YES
11843
11844  042442  005746                           TST     -(SP)           ;RESET SP TO ERROR VALUE
```

```
11845  042444  010603                          MOV     SP,R3           ;WAS SP
11846  042446  010504                          MOV     R5,R4
11847  042450  005744                          TST     -(R4)           ;S/B SP
11848  042452  104003                  7$:     ERROR   3               ;JSR FAILED TO PUSH SP
11849
11850  042454  010506                  8$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11851
11852                                  ;;**********************************************************************
11853                                  ;*TEST 621       JSR MODE 5 TEST
11854                                  ;;**********************************************************************
11855  042456                          T5T621:
11856  042456  000004                          SCOPE                   ;CALL THE SCOPE L. P UTILITY
11857  042460  012700  000621                  MOV     #621,R0         ;;LOAD R0 WITH TEST NUMBER
11858  042464  013701  042506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11859  042470  010605                          MOV     SP,R5           ;SAVE THE SP
11860  042472  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11861  042476  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11862  042500  012702  042536                  MOV     #7$,R2          ;DEST ADDR = [7$ - 2]
11863  042504  000257                          CCC                     ;SCOPE SYNC
11864
11865  042506  004452                  2$:     JSR     R4,@-(R2)       ;TEST THE JSR - GO TO 4$
11866
11867  042510  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11868
11869  042512  005726                  4$:     TST     (SP)+           ;RESET SP
11870  042514  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11871  042516  001411                          BEQ     T5T622          ;;BR IF YES
11872
11873  042520  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11874  042522  010603                          MOV     SP,R3           ;WAS SP
11875  042524  010504                          MOV     R5,R4
11876  042526  005744                          TST     -(R4)           ;S/B SP
11877  042530  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11878  042532  000402                          BR      6$              ;GO EXIT
11879
11880  042534  042512                          4$                      ;CONTAINS JUMP ADDRESS
11881  042536  104005                  7$:     ERROR   5               ;JSR EXECUTED LIKE A MODE 1 OR 2
11882
11883  042540  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11884
11885                                  ;;**********************************************************************
11886                                  ;*TEST 622       JSR MODE 6 TEST
11887                                  ;;**********************************************************************
11888  042542                          T5T622:
11889  042542  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11890  042544  012700  000622                  MOV     #622,R0         ;;LOAD R0 WITH TEST NUMBER
11891  042550  013701  042572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11892  042554  010605                          MOV     SP,R5           ;SAVE THE SP
11893  042556  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11894  042562  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11895  042564  012702  042576                  MOV     #3$,R2          ;[R2] = BASE DEST ADDR
11896  042570  000257                          CCC                     ;SCOPE SYNC
11897
11898  042572  004462  000002          2$:     JSR     R4,4$-3$(R2)    ;TEST THE JSR - GO TO 4$
11899
11900  042576  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC OR INDEX FAILED
```

```
11901
11902  042600  005726                  4$:     TST     (SP)+           ;RESET SP
11903  042602  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11904  042604  001406                          BEQ     TST623          ;;BR IF YES
11905
11906  042606  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11907  042610  010603                          MOV     SP,R3           ;WAS SP
11908  042612  010504                          MOV     R5,R4
11909  042614  005744                          TST     -(R4)           ;S/B SP
11910  042616  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11911  042620  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
11912
11913                                  ;;**********************************************************************
11914                                  ;*TEST 623       JSR MODE 7 TEST
11915                                  ;;**********************************************************************
11916  042622                          TST623:
11917  042622  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11918  042624  012700  000623                  MOV     #623,R0         ;;LOAD R0 WITH TEST NUMBER
11919  042630  013701  042652                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11920  042634  010605                          MOV     SP,R5           ;SAVE THE SP
11921  042636  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
11922  042642  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11923  042644  012702  042656                  MOV     #3$,R2          ;BASE DEST ADDR = 3$
11924  042650  000257                          CCC                     ;SCOPE SYNC
11925
11926  042652  004472  000024          2$:     JSR     R4,@7$-3$(R2)   ;TEST THE JSR - GO TO 4$ VIA 7$
11927
11928  042656  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11929                                                                  ;OR THE INDEX FAILED
11930
11931  042660  005726                  4$:     TST     (SP)+           ;RESET SP
11932  042662  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11933  042664  001411                          BEQ     TST624          ;;BR IF YES
11934
11935  042666  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11936  042670  010603                          MOV     SP,R3           ;WAS SP
11937  042672  010504                          MOV     R5,R4
11938  042674  005744                          TST     -(R4)           ;S/B SP
11939  042676  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11940  042700  000402                          BR      6$              ;SKIP TO EXIT
11941
11942  042702  042660                  7$:     4$                      ;CONTAINS JUMP ADDR
11943  042704  104005                          ERROR   5               ;JSR WORKED LIKE A MODE 1 OR 2
11944
11945  042706  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11946
11947                                  ;;**********************************************************************
11948                                  ;*TEST 624       SOB TEST, [R] = 1, NO BRANCH
11949                                  ;;**********************************************************************
11950  042710                          TST624:
11951  042710  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11952  042712  012700  000624                  MOV     #624,R0         ;;LOAD R0 WITH TEST NUMBER
11953  042716  013701  042736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11954  042722  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
11955  042726  000402                          BR      2$-2            ;GO DO THE SOB
11956
```

```
11957  042730  104006                3$:     ERROR   6               ;SOB SHOULDN'T HAVE BRANCHED HERE
11958  042732  000402                        BR      TST625          ;;GO TO SCOPE CALL
11959
11960  042734  000257                        CCC                     ;SYNC INSTR.
11961  042736  077204                2$:     SOB     R2,3$           ;TEST THE SOB
11962
11963                                ;;**************************************************************
11964                                ;*TEST 625      SOB TEST, [R] = 5, 6 ^CH 4 TIMES
11965                                ;;**************************************************************
11966  042740                        TST625:
11967  042740  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11968  042742  012700  000625                MOV     #625,R0         ;;LOAD R0 WITH TEST NUMBER
11969  042746  013701  043000                MOV     @#SOB2,R1       ;GET COPY OF TEST INSTRUCTION WORD
11970  042752  012702  000005                MOV     #5,R2           ;SET SOB COUNTER = 5
11971  042756  012705  177773                MOV     #-5,R5          ;SET UP R5 TO COUNT 5 BRANCHES
11972  042762  000405                        BR      SOB2-2          ;GO DO THE SOB
11973
11974  042764  000474                SOB1:   BR      SOB3            ;USED BY LAST SOB TEST TO TEST MAX OFFSET
11975  042766  000240                        NOP                     ;OFFSET ADJUSTMENT
11976  042770  000240                        NOP
11977
11978  042772  005205                SOB5:   INC     R5              ;COUNT ONE BRANCH
11979  042774  001406                        BEQ     SOBERR          ;BR IF TOO MANY LOOPS BY SOB
11980
11981  042776  000257                        CCC                     ;SCOPE SYNC
11982  043000  077204                SOB2:   SOB     R2,SOB5         ;TEST THE SOB
11983  043002  005702                        TST     R2              ;R2 SHOULD CONTAIN 0
11984  043004  001403                        BEQ     TST626          ;;BR IF IT DOES
11985
11986  043006  104006                        ERROR   6               ;SOB COUNTER NOT ZERO
11987  043010  000401                        BR      TST626          ;;GO TO SCOPE CALL
11988  043012  104006                SOBERR: ERROR   6               ;SOB MADE TOO MANY BRANCHES
11989
11990                                ;;**************************************************************
11991                                ;*TEST 626      SOB TEST, [R] = 1, FLAGS = 1111
11992                                ;;**************************************************************
11993  043014                        TST626:
11994  043014  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11995  043016  012700  000626                MOV     #626,R0         ;;LOAD R0 WITH TEST NUMBER
11996  043022  013701  043034                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11997  043026  012702  000001                MOV     #1,R2           ;SET SOB COUNTER = 1
11998  043032  000277                        SCC                     ;MAKE N:C = 1111
11999
12000  043034  077202                2$:     SOB     R2,2$-2         ;TEST THE SOB
12001
12002  043036  103003                        BCC     3$              ;BR IF C = 0
12003  043040  102002                        BVC     3$              ;BR IF V = 0
12004  043042  001001                        BNE     3$              ;BR IF Z = 0
12005  043044  100401                        BMI     TST627          ;;BR IF N = 1
12006
12007  043046  104006                3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
12008
12009                                ;;**************************************************************
12010                                ;*TEST 627      SOB TEST, [R] = 1, FLAGS = 0000
12011                                ;;**************************************************************
12012  043050                        TST627:
```

```
12013  043050  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12014  043052  012700  000627                  MOV     #627,R0         ;;LOAD R0 WITH TEST NUMBER
12015  043056  013701  043070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12016  043062  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
12017  043066  000257                          CCC                     ;MAKE N:C = 0000
12018
12019  043070  077202                  2$:     SOB     R2,2$-2         ;TEST THE SOB
12020
12021  043072  103403                          BCS     3$              ;BR IF C = 1
12022  043074  102402                          BVS     3$              ;BR IF V = 1
12023  043076  001401                          BEQ     3$              ;BR IF Z = 1
12024  043100  100001                          BPL     TST630          ;;BR IF N = 0
12025
12026  043102  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12027
12028                                  ;;*****************************************************************
12029                                  ;*TEST 630      SOB TEST, [R] = 5, FLAGS = 1111
12030                                  ;;*****************************************************************
12031  043104                          TST630:
12032  043104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12033  043106  012700  000630                  MOV     #630,R0         ;;LOAD R0 WITH TEST NUMBER
12034  043112  013701  043124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12035  043116  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12036  043122  000277                          SCC                     ;MAKE N:C = 1111
12037
12038  043124  077201                  2$:     SOB     R2,2$           ;TEST THE SOB
12039
12040  043126  103003                          BCC     3$              ;BR IF C = 0
12041  043130  102002                          BVC     3$              ;BR IF V = 0
12042  043132  001001                          BNE     3$              ;BR IF Z = 0
12043  043134  100401                          BMI     TST631          ;;BR IF N = 1
12044
12045  043136  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
12046
12047                                  ;;*****************************************************************
12048                                  ;*TEST 631      SOB TEST, [R] = 5, FLAGS = 0000
12049                                  ;;*****************************************************************
12050  043140                          TST631:
12051  043140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12052  043142  012700  000631                  MOV     #631,R0         ;;LOAD R0 WITH TEST NUMBER
12053  043146  013701  043160                  MOV     @#SOB4,R1       ;GET COPY OF TEST INSTRUCTION WORD
12054  043152  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12055  043156  000257                  SOB3:   CCC                     ;MAKE N:C = 0000
12056
12057  043160  077277                  SOB4:   SOB     R2,SOB1         ;TEST THE SOB
12058
12059  043162  103403                          BCS     3$              ;BR IF C = 1
12060  043164  102402                          BVS     3$              ;BR IF V = 1
12061  043166  001401                          BEQ     3$              ;BR IF Z = 1
12062  043170  100001                          BPL     TST632          ;;BR IF N = 0
12063
12064  043172  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12065
12066                                  ;;*****************************************************************
12067                                  ;*TEST 632      RTS TEST - N:C = 0000
12068                                  ;;*****************************************************************
```

```
12069  043174                          TST632:
12070  043174  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12071  043176  012700  000632          MOV     #632,R0         ;;LOAD R0 WITH TEST NUMBER
12072  043202  013701  043234          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12073  043206  010605                  MOV     SP,R5           ;SAVE THE SP
12074  043210  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12075  043214  012704  177777  1$:     MOV     #-1,R4          ;R3 SHOULD GET 177777
12076  043220  010506                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12077  043222  012703  043242          MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
12078  043226  012746  177777          MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 WITH 177777
12079  043232  000257                  CCC                     ;N:C = 0000
12080
12081  043234  000203          2$:     RTS     R3              ;TEST THE RTS - GO TO 4$
12082
12083  043236  104005          3$:     ERROR   5               ;RTS FAILED TO LOAD THE PC
12084  043240  000415                  BR      10$             ;GO TO EXIT - SCHOOLS OUT
12085
12086  043242  100403          4$:     BMI     5$              ;N:C = 0000 ?
12087  043244  001402                  BEQ     5$
12088  043246  102401                  BVS     5$
12089  043250  103001                  BCC     6$
12090
12091  043252  104005          5$:     ERROR   5               ;RTS ALTERED CODES - CLEARED ONE
12092
12093  043254  020403          6$:     CMP     R4,R3           ;DID R3 GET LOADED FROM STACK ?
12094  043256  001401                  BEQ     8$              ;BR IF YES
12095
12096  043260  104002          7$:     ERROR   2               ;RTS FAILED TO LOAD REG
12097
12098  043262  020506          8$:     CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
12099  043264  001404                  BEQ     TST633          ;;BR IF YES
12100
12101  043266  010504                  MOV     R5,R4           ;[R4] = S / B SP
12102  043270  010603                  MOV     SP,R3           ;[R3] = WAS SP
12103  043272  104003          9$:     ERROR   3               ;RTS FAILED TO POP SP
12104
12105  043274  010506          10$:    MOV     R5,SP           ;FIX THE SP
12106
12107                                  ;;*****************************************************************
12108                                  ;*TEST 633      RTT TEST - N:C = 1111
12109                                  ;;*****************************************************************
12110  043276                          TST633:
12111  043276  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12112  043300  012700  000633          MOV     #633,R0         ;;LOAD R0 WITH TEST NUMBER
12113  043304  013701  043346          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12114  043310  012702  177776          MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12115  043314  010605                  MOV     SP,R5           ;SAVE THE SP
12116  043316  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12117  043322  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12118  043324  012704  000340          MOV     #340,R4         ;[R4] = S / B PSW AT HTIS POINT
12119  043330  012746  000340          MOV     #340,-(SP)      ;NEW PSW S / B = 340
12120  043334  012746  043354          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12121  043340  005037  177776          CLR     @#PSW           ;CLEAR THE PSW
12122  043344  000277                  SCC                     ;N:C = 1111
12123
12124  043346  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
```

```
12125
12126  043350  104005                  3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12127  043352  000412                          BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12128
12129  043354  013703  177776          4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12130  043360  020403                          CMP     R4,R3           ;WAS PSW = 340 ?
12131  043362  001401                          BEQ     6$              ;BR IF IT WAS
12132
12133  043364  104001                  5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12134
12135  043366  020506                  6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12136  043370  001404                          BEQ     TST634          ;;BR IF YES
12137
12138  043372  010504                          MOV     R5,R4           ;[R4] = S / B SP
12139  043374  010603                          MOV     SP,R3           ;[R3] = WAS SP
12140  043376  104003                  7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12141
12142  043400  010506                  8$:     MOV     R5,SP           ;FIX THE SP
12143
12144                                  ;;**********************************************************
12145                                  ;*TEST 634       RTT TEST - N:C = 0000
12146                                  ;;**********************************************************
12147  043402                          TST634:
12148  043402  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12149  043404  012700  000634                  MOV     #634,R0         ;;LOAD R0 WITH TEST NUMBER
12150  043410  013701  043454                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12151  043414  012702  177776                  MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12152  043420  010605                          MOV     SP,R5           ;SAVE THE SP
12153  043422  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12154  043426  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12155  043430  012704  000017                  MOV     #017,R4         ;[R4] = S / B PSW AT THIS POINT
12156  043434  012746  000017                  MOV     #017,-(SP)      ;NEW PSW S / B = 017
12157  043440  012746  043462                  MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12158  043444  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW] = 340
12159  043452  000257                          CCC                     ;N:C = 0000
12160
12161  043454  000006                  2$:     RTT                     ;TEST THE RTT - GO TO 4$
12162
12163  043456  104005                  3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12164  043460  000412                          BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12165
12166  043462  013703  177776          4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12167  043466  020403                          CMP     R4,R3           ;WAS PSW = 017 ?
12168  043470  001401                          BEQ     6$              ;BR IF IT WAS
12169
12170  043472  104001                  5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12171
12172  043474  020506                  6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12173  043476  001404                          BEQ     TST635          ;;BR IF YES
12174
12175  043500  010504                          MOV     R5,R4           ;[R4] = S / B SP
12176  043502  010603                          MOV     SP,R3           ;[R3] = WAS SP
12177  043504  104003                  7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12178
12179  043506  010506                  8$:     MOV     R5,SP           ;FIX THE SP
12180
```

```
12181                                   ;;*****************************************************************
12182                                   ;*TEST 635      MARK INSTRUCTION TEST - N:C=0000
12183                                   ;;*****************************************************************
12184  043510                           TST635:
12185  043510  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12186  043512  012700  000635                   MOV     #635,R0         ;;LOAD R0 WITH TEST NUMBER
12187  043516  013701  043542                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12188  043522  010602                           MOV     SP,R2           ;SAVE SP
12189  043524  012704  125252                   MOV     #125252,R4      ;[R5] SHOULD BE 125252
12190  043530  012705  043572                   MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12191  043534  010437  043556                   MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12192  043540  000257                           CCC                     ;N:C=0000
12193
12194  043542  006405                   2$:     MARK+5                  ;TEST THE MARK
12195
12196  043544  010637  001074                   MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12197  043550  010206                           MOV     R2,SP           ;RESET SP
12198  043552  104005                   3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12199
12200  043554  000444                           BR      TST636          ;;GO TO SCOPE EXIT
12201
12202  043556  125252                   6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12203
12204  043560  010637  001074                   MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12205  043564  010206                           MOV     R2,SP           ;RESET SP
12206  043566  104005                   5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12207
12208  043570  000436                           BR      TST636          ;;GO TO SCOPE EXIT
12209
12210  043572  100403                   4$:     BMI     10$             ;N:C=0000?
12211  043574  001402                           BEQ     10$
12212  043576  102401                           BVS     10$
12213  043600  103011                           BCC     8$
12214
12215  043602  013703  177776           10$:    MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12216  043606  010637  001074                   MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12217  043612  010206                           MOV     R2,SP           ;RESET SP
12218  043614  012702  177776                   MOV     #PSW,R2         ;DEST=PSW
12219  043620  104007                   7$:     ERROR   7               ;MARK SET A FLAG
12220  043622  000421                           BR      TST636          ;;GO TO SCOPE EXIT
12221
12222  043624  020627  043560           8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12223  043630  001406                           BEQ     11$             ;BR IF YES
12224  043632  010603                           MOV     SP,R3           ;PUT BAD SP IN R3
12225  043634  012704  043560                   MOV     #6$+2,R4        ;S/B SP
12226  043640  010206                           MOV     R2,SP           ;RESET SP
12227  043642  104003                   9$:     ERROR   3               ;MARK FAILED TO RESET SP
12228
12229  043644  000410                           BR      TST636          ;;GO TO SCOPE EXIT
12230
12231  043646  020504                   11$:    CMP     R5,R4           ;DID MARK RESTORE OLD R5
12232  043650  001405                           BEQ     12$             ;BR IF YES
12233
12234  043652  010637  001074                   MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12235  043656  010503                           MOV     R5,R3           ;WAS DEST
12236  043660  010206                           MOV     R2,SP           ;RESET SP
```

```
12237  043662  104004                  ERROR   4               ;MARK FAILED TO RESET RS
12238
12239  043664  010206          12$:    MOV     R2,SP           ;RESET SP
12240
12241                          ;;**************************************************************
12242                          ;TEST 636       MARK INSTRUCTION TEST - N:C=1111
12243                          ;;**************************************************************
12244  043666                  TST636:
12245  043666  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12246  043670  012700  000636          MOV     #636,R0         ;;LOAD R0 WITH TEST NUMBER
12247  043674  013701  043720          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12248  043700  010602                  MOV     SP,R2           ;SAVE SP
12249  043702  012704  125252          MOV     #125252,R4      ;[R5] SHOULD BE 125252
12250  043706  012705  043750          MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12251  043712  010437  043734          MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12252  043716  000277                  SCC                     ;N:C=1111
12253
12254  043720  006405          2$:     MARK+5                  ;TEST THE MARK
12255
12256  043722  010637  001074          MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12257  043726  010206                  MOV     R2,SP           ;RESET SP
12258  043730  104005          3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12259
12260  043732  000444                  BR      TST637          ;;GO TO SCOPE EXIT
12261
12262  043734  125252          6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12263
12264  043736  010637  001074          MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12265  043742  010206                  MOV     R2,SP           ;RESET SP
12266  043744  104005          5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12267
12268  043746  000436                  BR      TST637          ;;GO TO SCOPE EXIT
12269
12270  043750  100003          4$:     BPL     7$              ;N:C=1111
12271  043752  001002                  BNE     7$
12272  043754  102001                  BVC     7$
12273  043756  103411                  BCS     8$
12274
12275  043760  013703  177776  7$:     MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12276  043764  010637  001074          MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12277  043770  010206                  MOV     R2,SP           ;RESET SP
12278  043772  012702  177776          MOV     #PSW,R2         ;DEST=PSW
12279  043776  104007                  ERROR   7               ;MARK SET A FLAG
12280  044000  000421                  BR      TST637          ;;GO TO SCOPE EXIT
12281
12282  044002  020627  043736  8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12283  044006  001406                  BEQ     9$              ;BR IF YES
12284  044010  010603                  MOV     SP,R3           ;PUT BAD SP IN R3
12285  044012  012704  043736          MOV     #6$+2,R4        ;S/B SP
12286  044016  010206                  MOV     R2,SP           ;RESET SP
12287  044020  104003                  ERROR   3               ;MARK FAILED TO RESET SP
12288
12289  044022  000410                  BR      TST637          ;;GO TO SCOPE EXIT
12290
12291  044024  020504          9$:     CMP     R5,R4           ;DID MARK RESTORE OLD R5
12292  044026  001405                  BEQ     10$             ;BR IF YES
```

```
12293
12294  044030  010637  001074          MOV     SP,@#SREGS      ;SAVE BAD SP FOR PRINTING
12295  044034  010503                  MOV     R5,R3           ;WAS DEST
12296  044036  010206                  MOV     R2,SP           ;RESET SP
12297  044040  104004                  ERROR   4               ;MARK FAILED TO RESET R5
12298
12299  044042  010206          10$:    MOV     R2,SP           ;RESET SP
12300
12301                                  ;;*****************************************************************
12302                                  ;*TEST 637      BASIC LINE CLOCK RESPONSE TEST
12303                                  ;;*****************************************************************
12304  044044                          TST637:
12305  044044  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12306  044046  012700  000637          MOV     #637,R0         ;;LOAD R0 WITH TEST NUMBER
12307  044052  013701  044102          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12308  044056  010605                  MOV     SP,R5           ;SAVE SP
12309  044060  012702  177546          MOV     #LKCSR,R2       ;[R2] = LINE CLOCK ADDRESS
12310  044064  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12311  044070  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12312  044072  012737  044106  000004  MOV     #4$,@#4         ;GO TO 4$ IF BUS TIMEOUT
12313  044100  000257                  CCC                     ;SCOPE SYNC
12314
12315  044102  005712          2$:     TST     (R2)            ;REFERENCE LKCSR ADDR
12316
12317  044104  000404                  BR      6$              ;GO TO EXIT
12318
12319  044106  012737  061220  000004  4$:     MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12320  044114  104006          3$:     ERROR   6               ;LKCSR FAILED TO RESPOND
12321
12322  044116  010506          6$:     MOV     R5,SP           ;RESET SP
12323  044120  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12324
12325                                  ;;*****************************************************************
12326                                  ;*TEST 640      LINE CLOCK TEST - LKCSR BIT 7 SET
12327                                  ;;*****************************************************************
12328  044126                          TST640:
12329  044126  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12330  044130  012700  000640          MOV     #640,R0         ;;LOAD R0 WITH TEST NUMBER
12331  044134  013701  044152          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12332  044140  012702  177546          MOV     #LKCSR,R2       ;DEST ADDR = 177546
12333  044144  012704  000200          MOV     #200,R4         ;[LKCSR) S / B = 200
12334  044150  000257                  CCC                     ;SCOPE SYNC
12335
12336  044152  030412          2$:     BIT     R4,(R2)         ;TEST BIT 7 IN LKCSR
12337
12338  044154  001002                  BNE     TST641          ;;BR IF IT'S SET
12339
12340  044156  011203                  MOV     (R2),R3         ;GET WAS DATA
12341  044160  104001          3$:     ERROR   1               ;BIT 7 NOT SET IN LKCSR
12342
12343                                  ;;*****************************************************************
12344                                  ;*TEST 641      LINE CLOCK TEST - LKCSR BIT 6 CLEAR
12345                                  ;;*****************************************************************
12346  044162                          TST641:
12347  044162  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12348  044164  012700  000641          MOV     #641,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 12349  044170  013701  044206                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 12350  044174  012702  177546                 MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
 12351  044200  012704  000200                 MOV     #200,R4         ;(LKCSR) S / B = 200
 12352  044204  000257                         CCC                     ;SCOPE SYNC
 12353
 12354  044206  032712  000100         2$:    BIT     #100,(R2)       ;TEST BIT 6 IN LKCSR
 12355
 12356  044212  001402                         BEQ     TST642          ;;BR IF CLEAR
 12357
 12358  044214  011203                         MOV     (R2),R3         ;GET WAS DATA
 12359  044216  104001                 3$:     ERROR   1               ;BIT 6 (INTR. ENAB.) IN LKCSR WAS SET
 12360
 12361                                 ;;**************************************************************
 12362                                 ;*TEST 642      LINE CLOCK TEST - LKCSR BIT 6 SET
 12363                                 ;;**************************************************************
 12364  044220                         TST642:
 12365  044220  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 12366  044222  012700  000642                 MOV     #642,R0         ;;LOAD R0 WITH TEST NUMBER
 12367  044226  013701  044276                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 12368  044232  010605                         MOV     SP,R5           ;SAVE SP
 12369  044234  012702  177546                 MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
 12370  044240  012704  000300                 MOV     #300,R4         ;(LKCSR) S / B = 300
 12371  044244  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
 12372  044250  012737  044312  000100  1$:    MOV     #4$,@#100       ;SET UP LCLK VECTOR IN CASE LOGIC
 12373  044256  012737  000340  000102         MOV     #340,@#102      ;FAULT CAUSES ATL INTERRUPT
 12374  044264  010506                         MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 12375  044266  012737  000340  177776         MOV     #340,@#PSW      ;SET PRIORITY TO LEVEL 7
 12376  044274  000257                         CCC                     ;SCOPE SYNC
 12377
 12378  044276  052712  000100         2$:    BIS     #100,(R2)       ;SET BIT 6 IN LKCSR
 12379
 12380  044302  020412                         CMP     R4,(R2)         ;RESULT CORRECT?
 12381  044304  001402                         BEQ     4$              ;BR IF YES
 12382
 12383  044306  011203                         MOV     (R2),R3         ;GET WAS DATA
 12384  044310  104001                 3$:     ERROR   1               ;BIT 6 FAILED TO SET IN LKCSR
 12385
 12386  044312  042737  000102  000100  4$:    BIC     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLOCK VECTOR
 12387  044320  005037  000102                 CLR     @#102
 12388  044324  042712  000100                 BIC     #100,(R2)       ;TURN OF LINE CLK INTR. ENAB.
 12389  044330  010506                         MOV     R5,SP           ;RESET SP
 12390
 12391                                 ;;**************************************************************
 12392                                 ;*TEST 643      LINE CLK BASIC INTERRUPT TEST
 12393                                 ;;**************************************************************
 12394  044332                         TST643:
 12395  044332  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 12396  044334  012700  000643                 MOV     #643,R0         ;;LOAD R0 WITH TEST NUMBER
 12397  044340  013701  044406                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 12398  044344  010605                         MOV     SP,R5           ;SAVE SP
 12399  044346  012702  177546                 MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
 12400  044352  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
 12401  044356  010506                 1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 12402  044360  005004                         CLR     R4              ;INITIALIZE TIMER
 12403  044362  012737  044424  000100         MOV     #4$,@#100       ;SET UP LINE CLOCK VECTOR TO TO
 12404  044370  012737  000340  000102         MOV     #340,@#102      ;TO 4$ WITH PROCESSOR PRIORITY = 7
```

```
12405  044376  005012                          CLR     (R2)            ;CLEAR LKCSR
12406  044400  005037  177776                  CLR     @#PSW           ;SET PRIORITY TO LEVEL 000
12407  044404  000257                          CCC                     ;SCOPE SYNC
12408
12409  044406  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPT
12410
12411  044412  005304                          DEC     R4              ;WAIT FOR INTR - REPORT ERROR IF
12412  044414  001376                          BNE     .-2             ;R4 GOES TO 000000
12413
12414  044416  042712  000100                  BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12415  044422  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTERRUPT
12416
12417  044424  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12418  044430  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLK VECTOR
12419  044436  005037  000102                  CLR     @#102
12420  044442  010506                          MOV     R5,SP           ;RESET SP
12421  044444  005037  177776                  CLR     @#PSW           ;RESET PRIORITY TO LEVEL 0
12422
12423                                  ;;**********************************************************************
12424                                  ;*TEST 644      RESET TEST - N:C = 1111
12425                                  ;;**********************************************************************
12426  044450                          TST644:
12427  044450  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12428  044452  012700  000644                  MOV     #644,R0         ;;LOAD R0 WITH TEST NUMBER
12429  044456  013701  044510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12430  044462  012737  000001  001110          MOV     #1,@#STIMES     ;NO ITERATIONS ON THIS TEST
12431  044470  012702  177564                  MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12432  044474  012737  000340  177776          MOV     #340,@#PSW      ;MAKE PRTY. BITS ALL 1'S
12433  044502  052712  000004                  BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12434  044506  000277                          SCC                     ;N:C = 1111
12435
12436  044510  000005                  2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12437
12438  044512  013705  177776                  MOV     @#PSW,R5        ;SAVE THE PSW
12439  044516  032712  000004                  BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12440  044522  001403                          BEQ     4$              ;BR IF YES
12441
12442  044524  042712  000004                  BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12443  044530  104006                  3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12444
12445  044532  022705  000357          4$:     CMP     #357,R5         ;DID RESET ALTER THE PSW ??
12446  044536  001406                          BEQ     6$              ;BR IF NOT
12447
12448  044540  012704  000357                  MOV     #357,R4         ;[R4] = S/B PSW
12449  044544  010503                          MOV     R5,R3           ;[R3] = WAS PSW
12450  044546  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12451  044552  104001                  5$:     ERROR   1               ;RESET ALTERED THE PSW
12452
12453  044554  005037  177776          6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
12454  044560  042737  000004  177564          BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12455
12456                                  ;;**********************************************************************
12457                                  ;*TEST 645      RESET TEST - N:C = 0000
12458                                  ;;**********************************************************************
12459  044566                          TST645:
12460  044566  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
12461  044570  012700  000645          MOV     #645,R0         ;;LOAD R0 WITH TEST NUMBER
12462  044574  013701  044624          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12463  044600  012737  000001  001110  MOV     #1,@#STIMES     ;NO ITERATIONS ON THIS TEST
12464  044606  012702  177564          MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12465  044612  005037  177776          CLR     @#PSW           ;MAKE PRTY. BITS ALL 0'S
12466  044616  052712  000004          BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12467  044622  000257                  CCC                     ;N:C = 0000
12468
12469  044624  000005          2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12470
12471  044626  013705  177776          MOV     @#PSW,R5        ;SAVE THE PSW
12472  044632  032712  000004          BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12473  044636  001403                  BEQ     4$              ;BR IF YES
12474
12475  044640  042712  000004          BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12476  044644  104006          3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12477
12478  044646  022705  000000          CMP     #0,R5           ;DID RESET ALTER THE PSW ??
12479  044652  001406                  BEQ     6$              ;BR IF NOT
12480
12481  044654  012704  000357          MOV     #357,R4         ;[R4] = S/B PSW
12482  044660  010503                  MOV     R5,R3           ;[R3] = WAS PSW
12483  044662  012702  177776          MOV     #PSW,R2         ;DEST = PSW
12484  044666  104001          5$:     ERROR   1               ;RESET ALTERED THE PSW
12485
12486  044670  005037  177776  6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
12487  044674  042737  000004  177564  BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12488
12489                          ;;*****************************************************************
12490                          ;*TEST 646      WAIT INSTRUCTION TEST - [PSW] = 151
12491                          ;;*****************************************************************
12492  044702                  TST646:
12493  044702  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12494  044704  012700  000646          MOV     #646,R0         ;;LOAD R0 WITH TEST NUMBER
12495  044710  013701  044776          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12496  044714  010605                  MOV     SP,R5           ;SAVE THE SP
12497  044716  010737  001010          MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
12498  044722  012702  177564  1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12499  044726  012737  045014  000064  MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12500  044734  012737  000200  000066  MOV     #200,@#66       ;AT LEVEL 4
12501  044742  010506                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12502  044744  005012                  CLR     (R2)            ;INIT DL11 XCSR
12503  044746  005003                  CLR     R3              ;INIT TIMER
12504
12505  044750  105712          3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12506  044752  100403                  BMI     5$              ;BR IF YES
12507  044754  005303                  DEC     R3              ;COUNT THE TIMER
12508  044756  001374                  BNE     3$              ;BR IF NO TIMEOUT
12509  044760  000440                  BR      9$              ;GO REPORT TIMEOUT
12510
12511  044762  012737  000140  177776  5$:     MOV     #140,@#PSW      ;SET PSW PRTY BITS TO LEVEL 3
12512  044770  000277                  SCC                     ;N:C=1111
12513  044772  152712  000100          BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1001
12514
12515  044776  000001          2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12516
```

```
12517  045000  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
12518  045006  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12519  045010  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12520  045012  000424                          BR      8$              ;GO EXIT THIS TEST
12521
12522  045014  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12523  045020  022716  045000                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12524  045024  001402                          BEQ     6$              ;BR IF YES
12525
12526  045026  104006                          ERROR   6               ;WAIT NOT FETCHED PROPERLY
12527  045030  000415                          BR      8$              ;GO EXIT THE TEST
12528
12529  045032  022766  000151  000002  6$:     CMP     #151,2(SP)      ;DID "WAIT" ALTER THE PSW ??
12530  045040  001411                          BEQ     8$              ;BR IF YES
12531
12532  045042  012704  000151                  MOV     #151,R4         ;[R4] = S/B PSW
12533  045046  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
12534  045052  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12535  045056  104001                  7$:     ERROR   1               ;"WAIT" ALTERED THE PSW
12536  045060  000401                          BR      8$              ;GOT TO EXIT TEST
12537
12538  045062  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12539
12540  045064  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12541  045066  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
12542  045072  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
12543  045074  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12544  045102  005037  000066                  CLR     @#66
12545
12546                                  ;;*****************************************************************
12547                                  ;*TEST 647      WAIT INSTRUCTION TEST - (PSW) = 010
12548                                  ;;*****************************************************************
12549  045106                          TST647:
12550  045106  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12551  045110  012700  000647                  MOV     #647,R0         ;;LOAD R0 WITH TEST NUMBER
12552  045114  013701  045200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12553  045120  010605                          MOV     SP,R5           ;SAVE THE SP
12554  045122  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12555  045126  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12556  045132  012737  045216  000064          MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12557  045140  012737  000200  000066          MOV     #200,@#66       ;AT LEVEL 4
12558  045146  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12559  045150  005012                          CLR     (R2)            ;INIT DL11 XCSR
12560  045152  005003                          CLR     R3              ;INIT TIMER
12561
12562  045154  105712                  3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12563  045156  100403                          BMI     5$              ;BR IF YES
12564  045160  005303                          DEC     R3              ;COUNT THE TIMER
12565  045162  001374                          BNE     3$              ;BR IF NO TIMEOUT
12566  045164  000437                          BR      9$              ;GO REPORT TIMEOUT
12567
12568  045166  005037  177776          5$:     CLR     @#PSW           ;SET PSW PRTY BITS TO LEVEL 0
12569  045172  000257                          CCC                     ;N:C=0000
12570  045174  152712  000100                  BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1000
12571
12572  045200  000001                  2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
```

```
12573
12574  045202  012737  000340  177776        MOV     #340,@#PSW      ;LOCK OUT INTR
12575  045210  005012                        CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12576  045212  104006                        ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12577  045214  000424                        BR      8$              ;GO EXIT THIS TEST
12578
12579  045216  042712  000100        4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12580  045222  022716  045202                CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12581  045226  001402                        BEQ     6$              ;BR IF YES
12582
12583  045230  104006                        ERROR   6               ;WAIT NOT FETCHED PROPERLY
12584  045232  000415                        BR      8$              ;GO EXIT THE TEST
12585
12586  045234  022766  000010  000002  6$:     CMP     #010,2(SP)      ;DID "WAIT" ALTER THE PSW ??
12587  045242  001411                        BEQ     8$              ;BR IF NO
12588
12589  045244  012704  000010                MOV     #010,R4         ;[R4] = S/B PSW
12590  045250  016603  000002                MOV     2(SP),R3        ;[R3] = WAS PSW
12591  045254  012702  177776                MOV     #PSW,R2         ;DEST = PSW
12592  045260  104001                7$:     ERROR   1               ;"WAIT" ALTERED THE PSW
12593  045262  000401                        BR      8$              ;GOT TO EXIT TEST
12594
12595  045264  104006                9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12596
12597  045266  010506                8$:     MOV     R5,SP           ;RESET THE SP
12598  045270  005037  177776                CLR     @#PSW           ;CLEAR OUT THE PSW
12599  045274  005012                        CLR     (R2)            ;TURN OFF DL11 INTR.
12600  045276  012737  000066  000064        MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12601  045304  005037  000066                CLR     @#66
12602
12603                                ;;*****************************************************************
12604                                ;*TEST 650      BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
12605                                ;;*****************************************************************
12606  045310                        TST650:
12607  045310  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12608  045312  012700  000650                MOV     #650,R0         ;;LOAD R0 WITH TEST NUMBER
12609  045316  013701  045364                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12610  045322  010605                        MOV     SP,R5           ;SAVE THE SP
12611  045324  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12612  045330  012702  177546        1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12613  045334  012737  045402  000100        MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12614  045342  012737  000340  000102        MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12615  045350  010506                        MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12616  045352  005004                        CLR     R4              ;INITIALIZE R4 AS TIMER
12617  045354  012737  000040  177776        MOV     #40,@#PSW       ;SET CPU PRIORITY TO LEVEL 1
12618  045362  000257                        CCC                     ;SCOPE SYNC
12619
12620  045364  052712  000100        2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12621
12622  045370  005304                        DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12623  045372  001376                        BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12624
12625  045374  042712  000100                BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12626  045400  104006                3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 1
12627
12628  045402  042712  000100        4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
```

```
12629  045406  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12630  045414  005037  000102                  CLR     @#102
12631  045420  010506                          MOV     R5,SP           ;RESET THE SP
12632  045422  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12633
12634                                  ;;*****************************************************************
12635                                  ;*TEST 651      BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
12636                                  ;;*****************************************************************
12637  045426                          TST651:
12638  045426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12639  045430  012700  000651                  MOV     #651,R0         ;;LOAD R0 WITH TEST NUMBER
12640  045434  013701  045502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12641  045440  010605                          MOV     SP,R5           ;SAVE THE SP
12642  045442  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12643  045446  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12644  045452  012737  045520  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12645  045460  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12646  045466  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12647  045470  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12648  045472  012737  000100  177776          MOV     #100,@#PSW      ;SET CPU PRIORITY TO LEVEL 2
12649  045500  000257                          CCC                     ;SCOPE SYNC
12650
12651  045502  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12652
12653  045506  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12654  045510  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12655
12656  045512  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12657  045516  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 2
12658
12659  045520  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12660  045524  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12661  045532  005037  000102                  CLR     @#102
12662  045536  010506                          MOV     R5,SP           ;RESET THE SP
12663  045540  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12664
12665                                  ;;*****************************************************************
12666                                  ;*TEST 652      BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
12667                                  ;;*****************************************************************
12668  045544                          TST652:
12669  045544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12670  045546  012700  000652                  MOV     #652,R0         ;;LOAD R0 WITH TEST NUMBER
12671  045552  013701  045620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12672  045556  010605                          MOV     SP,R5           ;SAVE THE SP
12673  045560  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12674  045564  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12675  045570  012737  045636  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12676  045576  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12677  045604  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12678  045606  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12679  045610  012737  000140  177776          MOV     #140,@#PSW      ;SET CPU PRIORITY TO LEVEL 3
12680  045616  000257                          CCC                     ;SCOPE SYNC
12681
12682  045620  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12683
12684  045624  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
```

```
12685  045626  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12686
12687  045630  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12688  045634  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 3
12689
12690  045636  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12691  045642  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12692  045650  005037  000102                  CLR     @#102
12693  045654  010506                          MOV     R5,SP           ;RESET THE SP
12694  045656  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12695
12696                                  ;;**************************************************************
12697                                  ;*TEST 653      BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
12698                                  ;;**************************************************************
12699  045662                          TST653:
12700  045662  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12701  045664  012700  000653                  MOV     #653,R0         ;;LOAD R0 WITH TEST NUMBER
12702  045670  013701  045736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12703  045674  010605                          MOV     SP,R5           ;SAVE THE SP
12704  045676  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12705  045702  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12706  045706  012737  045754  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12707  045714  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12708  045722  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12709  045724  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12710  045726  012737  000200  177776          MOV     #200,@#PSW      ;SET CPU PRIORITY TO LEVEL 4
12711  045734  000257                          CCC                     ;SCOPE SYNC
12712
12713  045736  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12714
12715  045742  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12716  045744  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12717
12718  045746  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12719  045752  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 4
12720
12721  045754  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12722  045760  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12723  045766  005037  000102                  CLR     @#102
12724  045772  010506                          MOV     R5,SP           ;RESET THE SP
12725  045774  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12726
12727                                  ;;**************************************************************
12728                                  ;*TEST 654      BR PRIORITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12729                                  ;;**************************************************************
12730  046000                          TST654:
12731  046000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12732  046002  012700  000654                  MOV     #654,R0         ;;LOAD R0 WITH TEST NUMBER
12733  046006  013701  046054                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12734  046012  010605                          MOV     SP,R5           ;SAVE THE SP
12735  046014  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12736  046020  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12737  046024  012737  046072  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12738  046032  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12739  046040  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12740  046042  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
```

```
12741  046044  012737  000240  177776          MOV     #240,@#PSW      ;SET CPU PRIORITY TO LEVEL 5
12742  046052  000257                          CCC                     ;SCOPE SYNC
12743
12744  046054  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12745
12746  046060  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12747  046062  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12748
12749  046064  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12750  046070  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 5
12751
12752  046072  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12753  046076  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12754  046104  005037  000102                  CLR     @#102
12755  046110  010506                          MOV     R5,SP           ;RESET THE SP
12756  046112  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12757
12758                                  ;;**********************************************************************
12759                                  ;*TEST 655      BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12760                                  ;;**********************************************************************
12761  046116                  TST655:
12762  046116  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12763  046120  012700  000655                  MOV     #655,R0         ;;LOAD R0 WITH TEST NUMBER
12764  046124  013701  046204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12765                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT13
12766  046130  032737  020000  063234          BIT     #BIT13,@#PTLOC  ;BREAKPOINT HALT SET ??
12767  046136  001401                          BEQ     .+4             ;BR IF NOT
12768  046140  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
12769  046142  010605                          MOV     SP,R5           ;SAVE THE SP
12770  046144  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
12771  046150  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12772  046154  012737  046216  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12773  046162  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12774  046170  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12775  046172  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12776  046174  012737  000300  177776          MOV     #300,@#PSW      ;SET CPU PRIORITY TO LEVEL 6
12777  046202  000257                          CCC                     ;SCOPE SYNC
12778
12779  046204  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12780
12781  046210  005304                          DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12782  046212  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12783  046214  000403                          BR      6$              ;GO TO EXIT - ALL OK
12784
12785  046216  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12786  046222  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 6
12787
12788  046224  042712  000100          6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12789  046230  012737  000102  000100          MOV     #102,@#100      ;RESET THE TRAP CATCHER IN THE VECTOR
12790  046236  005037  000102                  CLR     @#102
12791  046242  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12792  046244  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12793
12794                                  ;;**********************************************************************
12795                                  ;*TEST 656      BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12796                                  ;;**********************************************************************
```

```
12797 046350                      TST656:
12798 046350 000004                       SCOPE                     ;CALL THE SCOPE LOOP UTILITY
12799 046352 012700 000656                MOV     #656,R0           ;;LOAD R0 WITH TEST NUMBER
12800 046356 013701 046324                MOV     @#2$,R1           ;LOAD R1 WITH TEST INSTRUCTION WORD
12801 046362 010605                       MOV     SP,R5             ;SAVE THE SP
12802 046364 010737 001010                MOV     PC,@#$LPERR       ;SET ERROR LOOP ADDRESS
12803 046370 012702 177564          1$:   MOV     #XCSR,R2          ;R2 POINTS TO DL11 XCSR
12804 046374 012737 046336 000064         MOV     #4$,@#64          ;IF INTR OCCURS - GO TO 4$
12805 046402 012737 000340 000066         MOV     #340,@#66         ;WITH CPU PRIORITY AT LEVEL 7
12806 046410 010506                       MOV     R5,SP             ;RESET SP FOR ERROR LOOP
12807 046412 005004                       CLR     R4                ;INITIALIZE R4 AS TIMER
12808 046414 012737 000340 177776         MOV     #340,@#PSW        ;SET CPU PRIORITY TO LEVEL 7
12809 046422 000257                       CCC                       ;SCOPE SYNC
12810
12811 046424 052712 000100          2$:   BIS     #100,(R2)         ;ENABLE INTERRUPTS
12812
12813 046430 005304                       DEC     R4                ;COUNT UNTIL [R4] = 000000 - THEN
12814 046432 001376                       BNE     .-2               ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12815 046434 000403                       BR      6$                ;GO TO EXIT - ALL OK
12816
12817 046436 042712 000100          4$:   BIC     #100,(R2)         ;TURN OFF THE INTR ENABLE
12818 046442 104006                 3$:   ERROR   6                 ;INTR OCCURRED WITH CPU AT LEVEL 7
12819
12820 046444 042712 000100          6$:   BIC     #100,(R2)         ;TURN OFF INTR ENABLE
12821 046450 012737 000066 000064         MOV     #66,@#64          ;RESET THE TRAP CATCHER IN THE VECTOR
12822 046456 005037 000066                CLR     @#66
12823 046462 010506                       MOV     R5,SP             ;RESET SP JUST IN CASE
12824 046464 005037 177776                CLR     @#PSW             ;SET CPU PRIORITY BACK TO LEVEL 0
12825
12826                             ;;**********.**********************************************************
12827                             ;TEST 657      "CLR  @#PSW" ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12828                             ;THIS TEST VERIFIES THAT IF A "BR" REQUEST IS PENDING WHEN A "CLR  @#PSW"
12829                             ;IS EXECUTED TO LOWER THE CPU PRIORITY, THE REQUEST IS GRANTED BEFORE
12830                             ;EXECUTION OF THE INSTRUCTION FOLLOWING THE "CLR"
12831                             ;;**********************************************************************
12832 046470                      TST657:
12833 046470 000004                       SCOPE                     ;CALL THE SCOPE LOOP UTILITY
12834 046472 012700 000657                MOV     #657,R0           ;;LOAD R0 WITH TEST NUMBER
12835 046476 013701 046470                MOV     @#2$,R1           ;LOAD R1 WITH TEST INSTRUCTION WORD
12836 046502 012702 177546                MOV     #LKCSR,R2         ;R2 POINTS TO LINE CLK CSR
12837 046506 010605                       MOV     SP,R5             ;SAVE THE SP
12838 046510 010737 001010                MOV     PC,@#$LPERR       ;SET ERROR LOOP ADDRESS
12839 046514 012737 046476 000100   1$:   MOV     #4$,@#100         ;SET UP LCLK VECTOR TO GO TO 4$
12840 046522 012737 000300 000102         MOV     #300,@#102
12841 046530 010506                       MOV     R5,SP             ;RESET THE SP FOR ERROR LOOPING
12842 046532 005004                       CLR     R4                ;INITIALIZE TIMER FO KW
12843 046534 005003                       CLR     R3                ;CLEAR SOFTWARE FLAG
12844 046536 012737 000340 177776         MOV     #340,@#PSW        ;LOCK OUT ALL INTRS
12845 046544 052712 000100                BIS     #100,(R2)         ;ENABLE LCLK INTRS
12846 046550 042712 000200                BIC     #200,(R2)         ;CLEAR LINE CLOCK READY
12847 046554 105712                 11$:  TSTB    (R2)              ;LCLK READY TO INTR ??
12848 046556 100403                       BMI     12$               ;BR IF YES
12849 046560 005304                       DEC     R4                ;COUNT THE TIMER
12850 046562 001374                       BNE     11$               ;BR IF NO TIMEOUT
12851 046564 000411                       BR      6$                ;GO REPORT TIMEOUT
12852 046566 000257                 12$:  CCC                       ;SCOPE SYNC
```

```
12853
12854  046470  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - LCLK SHOULD INTERRUPT
12855                                                                   ;BEFORE FETCHING NEXT INSTRUCTION
12856  046474  005103                          COM     R3              ;SHOULD NOT BE FETCHED
12857  046476  005012                  4$:     CLR     (R2)            ;DISABLE THE LCLK INTR
12858  046500  005703                          TST     R3              ;DID SOFTWARE FLAG GET SET ??
12859  046502  001404                          BEQ     8$              ;BR IF NOT - IT WORKED OK
12860  046504  104006                  3$:     ERROR   6               ;LCLK FAILED TO INTR ONTIME
12861  046506  000402                          BR      8$              ;GO EXIT
12862
12863  046510  005012                  6$:     CLR     (R2)            ;DISABLE  LCLK INTR
12864  046512  104006                  5$:     ERROR   6               ;LINE CLK TIMED OUT
12865
12866  046514  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12867  046516  012737  000102  000100          MOV     #102,@#100      ;RESTORE THE LINE CLK TRAPCATCHER
12868  046524  005037  000102                  CLR     @#102
12869
12870                                  ;;*****************************************************************
12871                                  ;*TEST 660      "BR6 VS BR4" PRIORITY ARBITRATION TEST
12872                                  ;THIS TEST VERIFIES THAT IF BOTH A "BR4" AND A "BR6" REQUEST ARE
12873                                  ;PENDING WHEN THE CPU PRIORITY IS LOWERED TO ALLOW INTRS. THAT "BR6"
12874                                  ;REQUEST IS GRANTED FIRST EVEN THOUGH THE "BR4" REQUEST MAY HAVE
12875                                  ;OCCURRED FIRST
12876                                  ;;*****************************************************************
12877  046530                          TST660:
12878  046530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12879  046532  012700  000660                  MOV     #660,R0         ;;LOAD R0 WITH TEST NUMBER
12880  046536  013701  046676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12881  046542  010605                          MOV     SP,R5           ;SAVE THE SP
12882  046544  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
12883  046550  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12884  046554  012703  177564                  MOV     #XCSR,R3        ;R3 POINTS TO DL11 XCSR
12885  046560  012737  046706  000100          MOV     #4$,@#100       ;SET UP THE LCLK VECTOR - GO TO 4$
12886  046566  012737  000300  000102          MOV     #300,@#102
12887  046574  012737  046740  000064          MOV     #8$,@#64        ;SET UP THE DL11 VECTOR - GO TO 8$
12888  046602  012737  000200  000066          MOV     #200,@#66
12889  046610  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12890  046612  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12891  046620  005037  063312                  CLR     @#XBUF0         ;INIT TIMER
12892  046624  005037  063316                  CLR     @#XBUF1         ;CLEAR DL11 INTR FLAG
12893  046630  005004                          CLR     R4              ;INIT TIMER
12894  046632  052713  000100                  BIS     #100,(R3)       ;ENABLE DL11 XMIT INTR
12895  046636  105713                  11$:    TSTB    (R3)            ;XMIT READY SET ??
12896  046640  100403                          BMI     12$             ;BR IF YES
12897  046642  005304                          DEC     R4              ;COUNT THE TIMER
12898  046644  001374                          BNE     11$             ;BR IF NO TIMEOUT
12899  046646  000443                          BR      5$              ;GO REPORT TIMEOUT FOR DL11
12900
12901  046650  005004                  12$:    CLR     R4              ;INIT THE TIMER AGAIN
12902  046652  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12903  046656  042712  000200                  BIC     #200,(R2)       ;CLEAR THE LINE CLOCK READY BIT
12904  046662  105712                  13$:    TSTB    (R2)            ;LCLK READY TO INTR
12905  046664  100403                          BMI     14$             ;BR IF YES
12906  046666  005304                          DEC     R4              ;COUNT THE TIMER
12907  046670  001374                          BNE     13$             ;BR IF NO TIMEOUT
12908  046672  000436                          BR      7$              ;GO REPORT LINE CLK TIMEOUT
```

```
12909  046674  000257                  14$:    CCC                     ;SCOPE SYNC
12910
12911  046676  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - KW SHOULD INTR FIRST
12912
12913  046702  005037  063312                  COM     @#MBUF0         ;SET SOFTWARE FLAG IF FETCHED
12914  046706  005013                  4$:     CLR     (R3)            ;DISABLE BOTH INTERRUPTS
12915  046710  005012                          CLR     (R2)
12916  046712  005737  063312                  TST     @#MBUF0         ;DID SOFTWARE FLAG GET SET ??
12917  046716  001402                          BEQ     6$              ;BR IF NOT
12918
12919  046720  104006                  3$:     ERROR   6               ;LINE CLK INTR OCCURRED TOO LATE
12920  046722  000425                          BR      9$              ;GO TO EXIT
12921
12922  046724  005737  063316          6$:     TST     @#MBUF1         ;DID DL11 SOFTWARE FLAG SET ??
12923  046730  001422                          BEQ     9$              ;BR IF NOT
12924
12925  046732  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12926  046734  104006                          ERROR   6               ;DL11 INTERRUPTED THE KW11
12927  046736  000417                          BR      9$              ;GO TO EXIT TEST
12928
12929  046740  005137  063316          8$:     COM     @#MBUF1         ;FLAG THE DL11 INTR
12930  046744  005013                          CLR     (R3)            ;DISABLE BOTH INTR ENABLES
12931  046746  005012                          CLR     (R2)
12932  046750  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12933  046752  104006                          ERROR   6               ;DL11 SHOULD NOT HAVE INTERRUPTED
12934  046754  000410                          BR      9$              ;GO EXIT TEST
12935
12936  046756  005012                  5$:     CLR     (R2)            ;DISABLE THE INTR ENABLES
12937  046760  005013                          CLR     (R3)
12938  046762  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12939  046764  104006                          ERROR   6               ;DL11 TIMEOUT
12940  046766  000403                          BR      9$              ;GO TO EXIT
12941
12942  046770  005012                  7$:     CLR     (R2)            ;DISABLE INTR ENABLES
12943  046772  005013                          CLR     (R3)
12944  046774  104006                          ERROR   6               ;KW11 TIMEOUT
12945
12946  046776  010506                  9$:     MOV     R5,SP           ;RESET THE SP
12947  047000  005037  177776                  CLR     @#PSW           ;RESET THE CPU PRIORITY
12948  047004  012737  000102  000100          MOV     #102,@#100      ;RESTORE LCLK VECTOR
12949  047012  005037  000102                  CLR     @#102
12950  047016  012737  000066  000064          MOV     #66,@#64        ;RESTORE THE DL11 XMIT VECTOR
12951  047024  005037  000066                  CLR     @#66
12952
12953                                  ; ***************************************************************
12954                                  ; ////////////COMBINED INSTRUCTION EXERCISER SECTION ////////////
12955                                  ; ***************************************************************
12956
12957                                  ;;*************************************************************
12958                                  ;;TEST 661       "BPT" TRAP LINKAGE TEST
12959                                  ;;*************************************************************
12960  047030                          TST661:
12961  047030  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12962  047032  012700  000661                  MOV     #661,R0         ;;LOAD R0 WITH TEST NUMBER
12963  047036  013701  047062                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12964  047042  010605                          MOV     SP,R5           ;SAVE THE SP
```

```
12965  047044  010737  001010                  MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
12966  047050  012737  047066  000014  1$:     MOV     #4$,@#14        ;GO TO 4$ ON "BPT" TRAP
12967  047056  010506                          MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12968  047060  000257                          CCC                     ;SCOPE SYNC
12969
12970  047062  000003                  2$:     BPT                     ;TEST THE "BPT" - GO TO 4$
12971
12972  047064  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
12973
12974  047066  010506                  4$:     MOV     R5,SP           ;RESET THE SP
12975  047070  012737  000016  000014          MOV     #16,@#14        ;RESTORE THE VECTOR
12976
12977                                  ;;*****************************************************************
12978                                  ;*TEST 662      RED ZONE OVERFLOW TEST - MOV R,-(SP)
12979                                  ;;*****************************************************************
12980  047076                          ^ST662:
12981  047076  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12982  047100  012700  000662                  MOV     #662,R0         ;;LOAD R0 WITH TEST NUMBER
12983  047104  013701  047144                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12984  047110  010605                          MOV     SP,R5           ;SAVE SP
12985  047112  013704  000004                  MOV     @#4,R4          ;SAVE T.O. VECTOR
12986  047116  013703  000336                  MOV     @#336,R3        ;SAVE VECTOR AT 336
12987  047122  012737  047162  000004          MOV     #4$,@#4 ;GO TO 4$ ON C/FLW
12988  047130  012737  125252  000336          MOV     #125252,@#336   ;INIT. [336]
12989  047136  012706  000340                  MOV     #340,SP         ;SET SP TO CAUSE RED ZONE TRAP
12990  047142  000257                          CCC                     ;SCOPE SYNC
12991
12992  047144  010046                  2$:     MOV     R0,-(SP)        ;FORCE RED ZONE TRAP - GO TO 4$
12993
12994  047146  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
12995  047152  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12996  047156  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
12997  047160  104005                  3$:     ERROR   5               ;MOV FAILED TO CAUSE TRAP
12998
12999  047162  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13000  047166  022706  000000                  CMP     #0,SP           ;[SP]=0?
13001  047172  001404                          BEQ     6$              ;BE IF YES
13002
13003  047174  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13004  047200  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13005  047202  104005                  5$:     ERROR   5               ;SP NOT BEING JAMMED TO 4
13006
13007  047204  022737  125252  000336  6$:     CMP     #125252,@#336   ;DID PUSH OCCUR IN YELLOW ZONE?
13008  047212  001404                          BEQ     8$              ;BR IF NOT
13009
13010  047214  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13011  047220  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13012  047222  104005                  7$:     ERROR   5               ;MOV PUSHED INTO YELLOW ZONE
13013
13014  047224  010337  000336          8$:     MOV     R3,@#336        ;RESTORE VECTOR 336
13015  047230  010506                          MOV     R5,SP           ;RESET SP
13016
13017                                  ;;*****************************************************************
13018                                  ;*TEST 663      YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
13019                                  ;;*****************************************************************
13020  047232                          ^ST663:
```

```
13021  047232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13022  047234  012700  000663                  MOV     #663,R0         ;;LOAD R0 WITH TEST NUMBER
13023  047240  013701  047272                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13024  047244  010605                          MOV     SP,R5           ;SAVE SP
13025  047246  012702  000376                  MOV     #376,R2         ;R2 POINTS TO STACK
13026  047252  013704  000004                  MOV     @#4,R4          ;SAVE T.O. VECTOR
13027  047256  012737  047310  000004          MOV     #4$,@#4 ;ON OVFLW - GO TO 4$
13028  047264  012706  000400                  MOV     #400,SP         ;SET SP TO CAUSE OVFLW
13029  047270  000257                          CCC                     ;SCOPE SYNC
13030
13031  047272  010046                  2$:     MOV     R0,-(SP)        ;FORCE STACK OVFLW - GO TO 4$
13032
13033  047274  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13034  047300  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13035  047304  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13036  047306  104005                  3$:     ERROR   5               ;STACK OVFLW FAILED TO TRAP
13037
13038  047310  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13039  047314  020012                          CMP     R0,(R2)         ;DID [R0] GET PUSHED?
13040  047316  001404                          BEQ     6$              ;BR IF YES
13041
13042  047320  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13043  047324  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13044  047326  104005                  5$:     ERROR   5               ;MOV FAILED TO PUSH IN YELLOW ZONE
13045
13046  047330  005706                  6$:     TST     SP              ;[SP]=0?
13047  047332  001004                          BNE     8$              ;BR IF NOT
13048
13049  047334  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13050  047340  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13051  047342  104005                  7$:     ERROR   5               ;RED ZONE INSTEAD OF YELLOW ZONE
13052
13053  047344  010506                  8$:     MOV     R5,SP           ;RESET SP
13054
13055                                  ;;**********************************************************************
13056                                  ;*TEST 664       YELLOW ZONE OVERFLOW TEST - (CMP  R0,-(SP))
13057                                  ;;**********************************************************************
13058  047346                          TST664:
13059  047346  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13060  047350  012700  000664                  MOV     #664,R0         ;;LOAD R0 WITH TEST NUMBER
13061  047354  013701  047402                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13062  047360  010605                          MOV     SP,R5           ;SAVE THE SP
13063  047362  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
13064  047366  012737  047406  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13065  047374  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13066  047400  000257                          CCC                     ;SCOPE SYNC
13067
13068  047402  020046                  2$:     CMP     R0,-(SP)        ;TEST THE CMP - NO TRAP SHOULD OCCUR
13069
13070  047404  000406                          BR      6$              ;GO TO EXIT TEST
13071
13072  047406  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13073  047412  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13074  047416  010506                          MOV     R5,SP           ;RESET THE SP
13075  047420  104005                  3$:     ERROR   5               ;CMP CAUSED OVERFLOW TRAP
13076
```

```
13077  047422  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13078  047426  010506                          MOV     R5,SP           ;RESET THE SP
13079
13080                                  ;;*****************************************************************
13081                                  ;*TEST 665      YELLOW ZONE OVERFLOW TEST - (BIT  R0,-(SP))
13082                                  ;;*****************************************************************
13083  047430                          TST665:
13084  047430  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13085  047432  012700  000665                 MOV     #665,R0         ;;LOAD R0 WITH TEST NUMBER
13086  047436  013701  047464                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13087  047442  010605                          MOV     SP,R5           ;SAVE THE SP
13088  047444  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13089  047450  012737  047470  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13090  047456  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13091  047462  000257                         CCC                     ;SCOPE SYNC
13092
13093  047464  030046                  2$:     BIT     R0,-(SP)        ;TEST THE BIT - NO TRAP SHOULD OCCUR
13094
13095  047466  000406                          BR      6$              ;GO TO EXIT TEST
13096
13097  047470  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13098  047474  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13099  047500  010506                          MOV     R5,SP           ;RESET THE SP
13100  047502  104005                  3$:     ERROR   5               ;BIT CAUSED OVERFLOW TRAP
13101
13102  047504  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13103  047510  010506                          MOV     R5,SP           ;RESET THE SP
13104
13105                                  ;;*****************************************************************
13106                                  ;*TEST 666      YELLOW ZONE OVERFLOW TEST - (TST  -(SP))
13107                                  ;;*****************************************************************
13108  047512                          TST666:
13109  047512  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13110  047514  012700  000666                 MOV     #666,R0         ;;LOAD R0 WITH TEST NUMBER
13111  047520  013701  047546                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13112  047524  010605                          MOV     SP,R5           ;SAVE THE SP
13113  047526  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13114  047532  012737  047552  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13115  047540  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13116  047544  000257                         CCC                     ;SCOPE SYNC
13117
13118  047546  005746                  2$:     TST     -(SP)           ;TEST THE TST - NO TRAP SHOULD OCCUR
13119
13120  047550  000406                          BR      6$              ;GO TO EXIT TEST
13121
13122  047552  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13123  047556  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13124  047562  010506                          MOV     R5,SP           ;RESET THE SP
13125  047564  104006                  3$:     ERROR   6               ;TST CAUSED OVERFLOW TRAP
13126
13127  047566  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13128  047572  010506                          MOV     R5,SP           ;RESET THE SP
13129
13130                                  ;;*****************************************************************
13131                                  ;*TEST 667      ODD ADDRESS ERROR TEST - SUB  RA,(RB) - (RB) = ODD
13132                                  ;;*****************************************************************
```

```
13133  047574                          TST667:
13134  047574  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13135  047576  012700  000667                  MOV     #667,R0         ;;LOAD R0 WITH TEST NUMBER
13136  047602  013701  047636                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13137  047606  010605                          MOV     SP,R5           ;SAVE SP
13138  047610  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13139  047614  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13140  047620  012737  047646  000004        MOV     #4$,@#4 ;ON ODD ADDR ERROR - GO TO 4$
13141  047626  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13142  047630  012702  000001                  MOV     #1,R2           ;R2 GETS ODD ADDRESS
13143  047634  000257                          CCC                     ;SCOPE SYNC
13144
13145  047636  160012                  2$:     SUB     R0,(R2)         ;FORCE ODD ADDR ERROR - GO TO 4$
13146
13147  047640  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13148  047644  104006                  3$:     ERROR   6               ;ODD ADDR FAILED TO TRAP
13149
13150  047646  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13151  047652  010506                          MOV     R5,SP           ;RESET SP
13152  047654  005037  000000                  CLR     @#0             ;CLR LOC. 0 JUST IN CASE
13153
13154                                  ;;**************************************************************
13155                                  ;*TEST 670      TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
13156                                  ;;**************************************************************
13157  047660                          T670:
13158  047660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13159  047662  012700  000670                  MOV     #670,R0         ;;LOAD R0 WITH TEST NUMBER
13160  047666  013701  047710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13161  047672  012702  063317                  MOV     #MBUF1+1,R2     ;DEST ADDR=MBUF1+1 (ODD)
13162  047676  012737  047760  000004        MOV     #4$,@#4 ;GO TO 4$ ON ODD TRAP
13163
13164  047704  010205                          MOV     R2,R5           ;[R5] = DEST. ADDR
13165  047706  000257                          CCC                     ;SCOPE SYNC
13166
13167  047710  105435                  2$:     NEGB    @(R5)+          ;TEST DM=3 TRAP
13168
13169  047712  104006                  3$:     ERROR   6               ;ODD TRAP NOT SPRUNG
13170
13171  047714  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = DEST. ADDR
13172  047720  013701  047726                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13173  047724  000257                          CCC                     ;SCOPE SYNC
13174
13175  047726  105455                  20$:    NEGB    @-(R5)          ;TEST DM=5 TRAP
13176
13177  047730  104006                  5$:     ERROR   6               ;ODD TRAP NOT SPRUNG
13178
13179  047732  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13180  047734  013701  047742                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13181  047740  000257                          CCC                     ;SCOPE SYNC
13182
13183  047742  105475  000000          21$:    NEGB    @0(R5)          ;TEST DM=7 TRAP
13184
13185  047746  104006                  7$:     ERROR   6               ;ODD TRAP NOT SPRUNG
13186
13187  047750  012737  061220  000004        MOV     #BERR,@#4       ;RESET T.O. VECTOR
13188  047756  000403                          BR      TST671          ;;GO TO SCOPE EXIT
```

```
13189
13190  047760  062716  000002          4$:     ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13191  047764  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13192
13193                                  ;;*********************************************************************
13194                                  ;*TEST 671       TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
13195                                  ;;*********************************************************************
13196  047766                          TST671:
13197  047766  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13198  047770  012700  000671                  MOV     #671,R0         ;;LOAD R0 WITH TEST NUMBER
13199  047774  013701  050016                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13200  [illegible]  012702  063317            MOV     #MBUF1+1,R2     ;[R2] = SOURCE ADDR. (ODD)
13201  050004  012737  050066  000004          MOV     #4$,@#4 ;GO TO 4$ ON TRAP
13202
13203  050012  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR.
13204  050014  000257                          CCC                     ;SCOPE SYNC
13205
13206  050016  113504                  2$:     MOVB    @(R5)+,R4       ;TEST SM=3
13207
13208  050020  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13209
13210  050022  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = SOURCE ADDR
13211  [illegible]  013701  050034            MOV     @#20$,R1        ;[R1] = TEST INSTR
13212  050032  000257                          CCC                     ;SCOPE SYNC
13213
13214  050034  115504                  20$:    MOVB    @-(R5),R4       ;TEST SM=5
13215
13216  050036  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13217  [illegible]  010205                     MOV     R2,R5           ;[R5] = SOURCE ADDR
13218  050042  013701  050050                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13219  050046  000257                          CCC                     ;SCOPE SYNC
13220
13221  050050  117504  000000          21$:    MOVB    @0(R5),R4       ;TEST SM=7
13222
13223  050054  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13224
13225  050056  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13226  050064  000403                          BR      TST672          ;;GO TO SCOPE EXIT
13227
13228  050066  062716  000002          4$:     ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
13229  050072  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13230
13231                                  ;;*********************************************************************
13232                                  ;*TEST 672       TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES
13233                                  ;;*********************************************************************
13234  050074                          TST672:
13235  [illegible]  000004                     SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13236  050076  012700  000672                  MOV     #672,R0         ;;LOAD R0 WITH TEST NUMBER
13237  050102  013701  050124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13238  050106  012702  050177                  MOV     #6$+3,R2        ;DEST ADDR = 6$+3 (ODD)
13239  050112  012737  050202  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13240
13241  050120  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13242  050122  000257                          CCC                     ;SCOPE SYNC
13243
13244  050124  000135                  2$:     JMP     @(R5)+          ;TEST JMP DM=3
```

```
13245
13246 050126 104006                  3$:   ERROR   6              ;ODA TRAP NOT SPRUNG IN ROM LOC 153
13247
13248 050130 012705 050177                 MOV     #6$+3,R5       ;[R5] = DEST ADDR
13249 050134 013701 050142                 MOV     @#20$,R1       ;[R1] = TEST INSTR
13250 050140 000257                        CCC                    ;SCOPE SYNC
13251
13252 050142 000155                  20$:  JMP     @-(R5)         ;TEST JMP DM=5
13253
13254 050144 104006                  5$:   ERROR   6              ;ODA TRAP NOT SPRUNG IN ROM LOC 155
13255
13256 050146 010205                        MOV     R2,R5          ;[R5] = DEST ADDR
13257 050150 013701 050156                 MOV     @#21$,R1       ;[R1] = TEST INSTR
13258 050154 000257                        CCC                    ;SCOPE SYNC
13259
13260 050156 000175 000000           21$:  JMP     @0(R5)         ;TEST JMP DM=7
13261
13262 050162 104006                  7$:   ERROR   6              ;ODA TRAP NOT SPRUNG
13263
13264 050164 012737 061220 000004          MOV     #BERR,@#4      ;RESET BUS T.O. VECTOR
13265 050172 000420                        BR      TST673         ;;GO TO SCOPE EXIT
13266
13267 050174 000000                  6$:   HALT                   ;CATASTOPHIC ERROR - [PC] QUESTIONABLE.
13268 050176 000000                        HALT                   ;RESTART PROGRAM - DO NOT CONTINUE.
13269 050200 000000                        HALT
13270
13271 050202 032716 000001           4$:   BIT     #1,(SP)        ;TRAP DUE TO ODD PC?
13272 050206 001003                        BNE     8$             ;BR IF YES
13273 050210 062716 000002                 ADD     #2,(SP)        ;MOV RETURN PC AROUND ERROR CALL
13274 050214 000002                        RTI                    ;RETURN TO NEXT SUB TEST
13275
13276 050216 011603                  8$:   MOV     (SP),R3        ;GET ODD PC OFF STACK INTO R3
13277 050220 062706 000004                 ADD     #4,SP          ;FIX SP
13278
13279 050224 104007                  9$:   ERROR   7              ;PC TRAPPED WITH ODD ADDRESS
13280
13281 050226 012737 061220 000004          MOV     #BERR,@#4      ;RESET T.O. VECTOR
13282
13283                                ;;*****************************************************************
13284                                ;*TEST 673      TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
13285                                ;;*****************************************************************
13286 050234                        TST673:
13287 050234 000004                        SCOPE                  ;CALL THE SCOPE LOOP UTILITY
13288 050236 012700 000673                 MOV     #673,R0        ;;LOAD R0 WITH TEST NUMBER
13289 050242 013701 050266                 MOV     @#2$,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13290 050246 012737 050400 000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13291 050254 010605                        MOV     SP,R5          ;SAVE SP
13292 050256 012702 000376                 MOV     #376,R2        ;USE R2 TO SET UP SP TO CAUSE TRAP
13293
13294 050262 010206                        MOV     R2,SP          ;SET UP SP TO CAUSE OVERFLOW
13295 050264 000257                        CCC                    ;SCOPE SYNC
13296
13297 050266 005016                  2$:   CLR     (SP)           ;TEST DM1 - SHOULD SPRING TRAP
13298
13299 050270 010637 001074                 MOV     SP,@#$REGS     ;SAVE BAD SP FOR PRINTING
13300 050274 010506                        MOV     R5,SP          ;RESET SP
```

```
13301  050276  104006                  3$:   ERROR   6               ;DM1 FAILED TO CAUSE OVERFLOW TRAP
13302
13303  050300  013701  050310                MOV     @#20$,R1        ;[R1] = TEST INSTR.
13304  050304  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13305  050306  000257                        CCC                     ;SCOPE SYNC
13306
13307  050310  005026                  20$:  CLR     (SP)+           ;TEST DM2 - SHOULD SPRING TRAP
13308
13309  050312  010637  001074                MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13310  050316  010506                        MOV     R5,SP           ;RESET SP
13311  050320  104006                  5$:   ERROR   6               ;DM2 FAILED TO CAUSE OVERFLOW TRAP
13312
13313  050322  013701  050332                MOV     @#21$,R1        ;[R1] = TEST INSTR.
13314  050326  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13315  050330  000257                        CCC                     ;SCOPE SYNC
13316
13317  050332  005046                  21$:  CLR     -(SP)           ;TEST DM4 - SHOULD SPRING TRAP
13318
13319  050334  010637  001074                MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13320  050340  010506                        MOV     R5,SP           ;RESET SP
13321  050342  104006                  7$:   ERROR   6               ;DM4 FAILED TO CAUSE OVERFLOW TRAP
13322
13323  050344  013701  050354                MOV     @#22$,R1        ;[R1] = TEST INSTR.
13324  050350  010206                        MOV     R2,SP           ;SET SP TO CAUSE ERROR
13325  050352  000257                        CCC                     ;SCOPE SYNC
13326
13327  050354  005066  000000          22$:  CLR     0(SP)           ;TEST DM6 - SHOULD SPRING TRAP
13328
13329  050360  010637  001074                MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13330  050364  010506                        MOV     R5,SP           ;RESET SP
13331  050366  104006                  9$:   ERROR   6               ;DM6 FAILED TO CAUSE OVERFLOW TRAP
13332
13333  050370  012737  061220  000004        MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13334  050376  000407                        BR      TST674          ;;GO TO SCOPE EXIT
13335
13336  050400  011604                  4$:   MOV     (SP),R4         ;GET RETURN PC OFF STACK
13337  050402  062704  000010                ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13338  050406  010506                        MOV     R5,SP           ;RESET SP
13339  050410  005046                        CLR     -(SP)           ;PUSH NEW PS ON STACK
13340  050412  010446                        MOV     R4,-(SP)        ;PUSH RETURN PC ON STACK
13341  050414  000002                        RTI                     ;RETURN TO NEXT SUB-TEST
13342
13343                                  ;;**********************************************************************
13344                                  ;*TEST 674      TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
13345                                  ;;**********************************************************************
13346  050416                          TST674:
13347  050416  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13348  050420  012700  000674                MOV     #674,R0         ;;LOAD R0 WITH TEST NUMBER
13349  050424  013701  050450                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13350  050430  012737  050562  000004        MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLW TRAP
13351  050436  010605                        MOV     SP,R5           ;SAVE SP
13352  050440  012702  000376                MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13353
13354  050444  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13355  050446  000257                        CCC                     ;SCOPE SYNC
13356
```

```
 13357  050450  010016                    2$:     MOV     R0,(SP)         ;TEST MOV DM1 - SHOULD SPRING TRAP
 13358
 13359  050452  010637  001074                    MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
 13360  050456  010506                    3$:     MOV     R5,SP           ;RESET SP
 13361  050460  104006                            ERROR   6               ;MOV DM1 FAILED TO SPRING TRAP
 13362
 13363  050462  013701  050472                    MOV     @#20$,R1        ;[R1] = TEST INSTR.
 13364  050466  010206                            MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
 13365  050470  000257                            CCC                     ;SCOPE SYNC
 13366
 13367  050472  010026                    20$:    MOV     R0,(SP)+        ;TEST MOV DM2 - SHOULD SPRING TRAP
 13368
 13369  050474  010637  001074                    MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
 13370  050500  010506                    4$:     MOV     R5,SP           ;RESET SP
 13371  050502  104006                            ERROR   6               ;MOV DM2 FAILED TO SPRING TRAP
 13372
 13373  050504  013701  050514                    MOV     @#21$,R1        ;[R1] = TEST INSTR.
 13374  050510  010206                            MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
 13375  050512  000257                            CCC                     ;SCOPE SYNC
 13376
 13377  050514  010046                    21$:    MOV     R0,-(SP)        ;TEST MOV DM4 - SHOULD SPRING TRAP
 13378
 13379  050516  010637  001074                    MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
 13380  050522  010506                    6$:     MOV     R5,SP           ;RESET SP
 13381  050524  104006                            ERROR   6               ;MOV DM4 FAILED TO SPRING TRAP
 13382
 13383  050526  013701  050536                    MOV     @#22$,R1        ;[R1] = TEST INSTR.
 13384  050532  010206                            MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
 13385  050534  000257                            CCC                     ;SCOPE SYNC
 13386
 13387  050536  010066  000000            22$:    MOV     R0,0(SP)        ;TEST MOV DM6 - SHOULD SPRING TRAP
 13388
 13389  050542  010637  001074                    MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
 13390  050546  010506                    9$:     MOV     R5,SP           ;RESET SP
 13391  050550  104006                            ERROR   6               ;MOV DM6 FAILED TO CAUSE OVFLW TRAP
 13392
 13393  050552  012737  061220  000004            MOV     #BERR,@#4       ;RESET T.O. VECTOR
 13394  050560  000407                            BR      TST675          ;;GO TO SCOPE EXIT
 13395
 13396  050562  011604                    4$:     MOV     (SP),R4         ;GET RETURN PC
 13397  050564  062704  000010                    ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
 13398  050570  010506                            MOV     R5,SP           ;RESET SP
 13399  050572  005046                            CLR     -(SP)           ;PUSH NEW PSW
 13400  050574  010446                            MOV     R4,-(SP)        ;PUSH RETURN PC
 13401  050576  000002                            RTI                     ;RETURN TO NEXT SUB-TEST
 13402
 13403                                    ;;**************************************************************
 13404                                    ;*TEST 675      TEST THAT JSR CAN CAUSE OVERFLOW TRAP
 13405                                    ;;**************************************************************
 13406  050600                            TST675:
 13407  050600  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 13408  050602  012700  000675                    MOV     #675,R0         ;;LOAD R0 WITH TEST NUMBER
 13409  050606  013701  050630                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 13410  050612  012737  050652  000004            MOV     #4$,@#4 ;GO TO 4$ ON OVERFLOW ERROR
 13411  050620  010605                            MOV     SP,R5           ;SAVE SP
 13412  050622  012706  000400                    MOV     #400,SP         ;SET THE SP TO CAUSE TRAP
```

```
13413  050626  000257                          CCC                     ;SCOPE SYNC
13414
13415  050630  004737  050656          2$:     JSR     PC,@#6$ ;TEST JSR - SHOULD SPRING TRAP
13416
13417  050634  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
13418  050640  010506                          MOV     R5,SP           ;RESET SP
13419  050642  104005                  3$:     ERROR   5               ;JSR PUSH DID NOT SPRING OVFL TRAP
13420
13421  050644  000410                          BR      8$              ;GO TO SCOPE EXIT
13422
13423  050646  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
13424  050652  010506                  4$:     MOV     R5,SP           ;RESET SP
13425  050654  000404                          BR      8$              ;GO EXIT TEST - ALL OK
13426
13427  050656  010637  001074          6$:     MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
13428  050662  010506                          MOV     R5,SP           ;RESET SP
13429  050664  104005                  5$:     ERROR   5               ;JSR PUSH FAILED TO SPRING OVFLW TRAP
13430
13431  050666  012737  061220  000004  8$:     MOV     #ERR,@#4        ;RESET BUS T.O. VECTOR
13432
13433                                  ;;*****************************************************************
13434                                  ;*TEST 676      TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
13435                                  ;;*****************************************************************
13436  050674                          TST676:
13437  050674  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13438  050676  012700  000676                  MOV     #676,R0         ;;LOAD R0 WITH TEST NUMBER
13439  050702  013701  050736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13440  050706  013704  000014                  MOV     @#14,R4         ;SAVE BREAK POINT TRAP VECTOR
13441  050712  010605                          MOV     SP,R5           ;SAVE SP
13442  050714  012737  050752  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13443  050722  012737  050756  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13444  050730  012706  000400                  MOV     #400,SP         ;SET UP SP TO CAUSE OVFLW ON 1ST PUSH
13445  050734  000257                          CCC                     ;SCOPE SYNC
13446
13447  050736  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13448
13449  050740  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
13450  050744  010506                          MOV     R5,SP           ;RESET SP
13451  050746  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
13452
13453  050750  000406                          BR      8$              ;GO TO SCOPE EXIT
13454
13455  050752  010506                  4$:     MOV     R5,SP           ;RESET SP
13456  050754  000404                          BR      8$              ;GO EXIT - ALL OK
13457
13458  050756  010637  001074          6$:     MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
13459  050762  010506                          MOV     R5,SP           ;RESET SP
13460  050764  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13461
13462  050766  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13463  050774  010437  000014                  MOV     R4,@#14
13464
13465                                  ;;*****************************************************************
13466                                  ;*TEST 677      TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
13467                                  ;;*****************************************************************
13468  051000                          TST677:
```

```
13469  051000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13470  051002  012700  000677                  MOV     #677,R0         ;;LOAD R0 WITH TEST NUMBER
13471  051006  013701  051042                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13472  051012  013704  000014                  MOV     @#14,R4         ;SAVE BPT VECTOR
13473  051016  010605                          MOV     SP,R5           ;SAVE SP
13474  051020  012737  051056  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLOW
13475  051026  012737  051062  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13476  051034  012706  000402                  MOV     #402,SP         ;SET SP TO CAUSE TRAP ON 2ND PUSH
13477  051040  000257                          CCC                     ;SCOPE SYNC
13478
13479  051042  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13480
13481  051044  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13482  051050  010506                          MOV     R5,SP           ;RESET SP
13483  051052  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
13484
13485  051054  000406                          BR      8$              ;GO TO SCOPE EXIT
13486
13487  051056  010506                  4$:     MOV     R5,SP           ;RESET SP
13488  051060  000404                          BR      8$              ;GO EXIT - ALL OK
13489
13490  051062  010637  001074          6$:     MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
13491  051066  010506                          MOV     R5,SP           ;RESET SP
13492  051070  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13493
13494  051072  012737  061220  000004  8$:     MOV     #8ERR,@#4       ;RESET VECTORS
13495  051100  010437  000014                  MOV     R4,@#14
13496
13497                                  ;;*****************************************************************
13498                                  ;*TEST 700      ILLEGAL INSTRUCTION TEST - JSR RN,%R
13499                                  ;;*****************************************************************
13500  051104                          1ST700:
13501  051104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13502  051106  012700  000700                  MOV     #700,R0         ;;LOAD R0 WITH TEST NUMBER
13503  051112  013701  051146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13504  051116  010605                          MOV     SP,R5           ;SAVE SP
13505  051120  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13506  051124  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13507  051130  012737  051156  000004          MOV     #4$,@#4 ;ILLEGAL INSTR. TRAP GOES TO 4$
13508  051136  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13509  051140  012702  051154                  MOV     #3$,R2          ;IN CASE JSR JUMPS TO [R2]
13510  051144  000257                          CCC                     ;SCOPE SYNC
13511
13512  051146  004302                  2$:     JSR     R3,R2           ;JSR MODE 0 FORCES TRAP - GO TO 4$
13513
13514  051150  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13515  051154  104005                  3$:     ERROR   5               ;JSR FAILED TO SPRING TRAP
13516
13517  051156  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR
13518  051162  010506                          MOV     R5,SP           ;RESET SP
13519
13520                                  ;;*****************************************************************
13521                                  ;*TEST 701      ILLEGAL INSTRUCTION TEST - JMP %R
13522                                  ;;*****************************************************************
13523  051164                          1ST701:
13524  051164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 13525  051166  012700  000701                  MOV     #701,R0         ;;LOAD R0 WITH TEST NUMBER
 13526  051172  013701  051226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 13527  051176  010605                          MOV     SP,R5           ;SAVE SP
 13528  051200  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
 13529  051204  013704  000004          1$:     MOV     @#4,R4          ;SAVE VECTOR POINTER AT LOC. 4
 13530  051210  012737  051236  000004          MOV     #4$,@#4 ;ON TRAP - GO TO 4$
 13531  051216  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 13532  051220  012702  051234                  MOV     #3$,R2          ;IN CASE IT JUMPS TO ADDR IN R4
 13533  051224  000257                          CCC                     ;SCOPE SYNC
 13534
 13535  051226  000102                  2$:     JMP     R2              ;JMP MODE 0 FORCES TRAP - GO TO 4$
 13536
 13537  051230  010437  000004                  MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC  4
 13538  051234  104005                  3$:     ERROR   5               ;ILLEGAL INSTR TRAP FAILED
 13539
 13540  051236  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
 13541  051242  010506                          MOV     R5,SP           ;RESET SP
 13542
 13543                                  ;;**********************************************************************
 13544                                  ;*TEST 702      BUS TIMEOUT TRAP TEST - TST (R)
 13545                                  ;;**********************************************************************
 13546  051244                          TST702:
 13547  051244  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 13548  051246  012700  000702                  MOV     #702,R0         ;;LOAD R0 WITH TEST NUMBER
 13549  051252  013701  051306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 13550  051256  010605                          MOV     SP,R5           ;SAVE SP
 13551  051260  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
 13552  051264  013704  000004          1$:     MOV     @#4,R4          ;SAVE ORIGINAL T.O. VECTOR POINTER
 13553  051270  012737  051316  000004          MOV     #4$,@#4 ;ON T.O. TRAP - GO TO 4$
 13554  051276  012702  160000                  MOV     #160000,R2      ;ADDRESS CAUSES T.O.
 13555  051302  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 13556  051304  000257                          CCC                     ;SCOPE SYNC
 13557
 13558  051306  005712                  2$:     TST     (R2)            ;FORCE T.O. TRAP - GO TO 4$
 13559
 13560  051310  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
 13561  051314  104005                  3$:     ERROR   5               ;TIMEOUT TRAP FAILED
 13562  051316  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
 13563  051322  010506                          MOV     R5,SP           ;RESET SP
 13564
 13565                                  ;;**********************************************************************
 13566                                  ;*TEST 703      "T" BIT TRAP TEST
 13567                                  ;;**********************************************************************
 13568  051324                          TST703:
 13569  051324  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 13570  051326  012700  000703                  MOV     #703,R0         ;;LOAD R0 WITH TEST NUMBER
 13571  051332  013701  051370                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 13572  051336  010605                          MOV     SP,R5           ;SAVE SP
 13573  051340  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
 13574  051344  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 13575  051346  012737  051376  000014          MOV     #4$,@#14        ;GO TO 4$ WHEN "T" TRAP SPRUNG
 13576  051354  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT ON STACK
 13577  051360  012746  051370                  MOV     #2$,-(SP)       ;SET UP NEW PC ON STACK
 13578  051364  000257                          CCC                     ;SCOPE SYNC
 13579  051366  000006                          RTT                     ;TURN ON "T" BIT - GO TO 2$
 13580
```

```
13581  051370  005700                  2$:     TST     R0              ;SPRING "T" BIT TRAP - GO TO 4$
13582
13583  051372  104005                  3$:     ERROR   5               ;NO "T" BIT TRAP OCCURRED
13584
13585  051374  000405                          BR      6$              ;GO EXIT
13586
13587  051376  032766  000020  000002  4$:     BIT     #20,2(SP)       ;"T" BIT SET IN OLD PSW?
13588  051404  001001                          BNE     6$              ;BR IF YES
13589
13590  051406  104001                  5$:     ERROR   1               ;#T# BIT NOT SAVED ON STACK
13591
13592  051410  012737  000016  000014  6$:     MOV     #16,@#14        ;RESTORE "T" BIT TRAP CATCHER
13593  051416  005037  000016                  CLR     @#16
13594  051422  010506                          MOV     R5,SP           ;RESET SP
13595
13596                                  ;;**************************************************************
13597                                  ;*TEST 704      TEST PUSH INTO PSW WITH [SP] = 000000
13598                                  ;THESE NEXT TWO TESTS VERIFY THAT A "RED ZONE" TRAP OCCURS IF A
13599                                  ;PUSH IS ATTEMPTED WITH THE [SP] INITIALLY EQUAL TO 000000,177572,
13600                                  ;;**************************************************************
13601  051424                          TST704:
13602  051424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13603  051426  012700  000704                  MOV     #704,R0         ;;LOAD R0 WITH TEST NUMBER
13604  051432  013701  051456                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13605  051436  010605                          MOV     SP,R5           ;SAVE THE SP
13606  051440  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13607  051444  012737  051500  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13610  051452  005006                          CLR     SP              ;MAKE SP = 000000
13609  051454  000257                          CCC                     ;SCOPE SYNC
13610
13611  051456  012746  007777          2$:     MOV     #7777,-(SP)     ;ATTEMPT PUSH INTO PSW - SHOULD CAUSE
13612                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
13613
13614  051462  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13615  051466  005004                          CLR     R4              ;[R4] = S / B SP
13616  051470  010603                          MOV     SP,R3           ;[R3] = WAS SP
13617  051472  010506                          MOV     R5,SP           ;RESET THE SP
13618  051474  104003                  3$:     ERROR   3               ;TRAP NOT SPRUNG
13619  051476  000414                          BR      TST705          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13620
13621  051500  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13622  051504  001406                          BEQ     6$              ;BR IF YES
13623
13624  051506  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13625  051512  005004                          CLR     R4              ;[R4]= S / B SP
13626  051514  010603                          MOV     SP,R3           ;[R3] = WAS SP
13627  051516  010506                          MOV     R5,SP           ;RESET THE SP
13628  051520  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13629
13630  051522  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
13631  051524  010437  000004                  MOV     R4,@#4          ;RESTORE BERR VECTOR
13632
13633                                  ;;**************************************************************
13634                                  ;*TEST 705      TEST PUSH INTO SR WITH [SP] = 177572
13635                                  ;;**************************************************************
13636  051530                          TST705:
```

```
13637  051530  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13638  051532  012700  000705                  MOV     #705,R0                 ;;LOAD R0 WITH TEST NUMBER
13639  051536  013701  051564                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
13640  051542  010605                          MOV     SP,R5                   ;SAVE THE SP
13641  051544  013704  000004                  MOV     @#4,R4                  ;SAVE THE BUS ERROR VECTOR
13642  051550  012737  051606  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13643  051556  012706  177572                  MOV     #177572,SP              ;MAKE SP=177572
13644  051562  000257                          CCC                             ;SCOPE SYNC
13645
13646  051564  012746  177777          2$:     MOV     #-1,-(SP)               ;ATTEMPT PUSH INTO SR - SHOULD CAUSE
13647                                                                          ;"RED ZONE" TRAP TO BE SPRUNG
13648
13649  051570  010437  000004                  MOV     R4,@#4                  ;RESTORE BUS ERROR VECTOR
13650  051574  005004                          CLR     R4                      ;[R4] = S / B SP
13651  051576  010603                          MOV     SP,R3                   ;[R3] = WAS SP
13652  051600  010506                          MOV     R5,SP                   ;RESET THE SP
13653  051602  104003          3$:     ERROR   3                               ;TRAP NOT SPRUNG
13654  051604  000414                          BR      TST706                  ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13655
13656  051606  022706  000000          4$:     CMP     #0,SP                   ;WAS IT A RED ZONE TRAP ?
13657  051612  001406                          BEQ     6$                      ;BR IF YES
13658
13659  051614  010437  000004                  MOV     R4,@#4                  ;RESTORE BUS ERROR VECTOR
13660  051620  005004                          CLR     R4                      ;[R4]= S / B SP
13661  051622  010603                          MOV     SP,R3                   ;[R3] = WAS SP
13662  051624  010506                          MOV     R5,SP                   ;RESET THE SP
13663  051626  104003          5$:     ERROR   3                               ;TRAP SPRUNG BUT NOT RED ZONE
13664
13665  051630  010506          6$:     MOV     R5,SP                   ;FIX UP THE SP
13666  051632  010437  000004                  MOV     R4,@#4                  ;RESTORE BUS ERROR VECTOR
13667
13668                                  ;;*****************************************************************
13669                                  ;*TEST 706       TEST PUSH INTO SLR WITH [SP] = 177776
13670                                  ;;*****************************************************************
13671  051636                          TST706:
13672  051636  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
13673  051640  012700  000706                  MOV     #706,R0                 ;;LOAD R0 WITH TEST NUMBER
13674  051644  013701  051672                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
13675  051650  010605                          MOV     SP,R5                   ;SAVE THE SP
13676  051652  013704  000004                  MOV     @#4,R4                  ;SAVE THE BUS ERROR VECTOR
13677  051656  012737  051714  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13678  051664  012706  177776                  MOV     #177776,SP              ;MAKE SP=177776
13679  051670  000257                          CCC                             ;SCOPE SYNC
13680
13681  051672  012746  000200          2$:     MOV     #200,-(SP)              ;ATTEMPT PUSH INTO SLR - SHOULD CAUSE
13682                                                                          ;"RED ZONE" TRAP TO BE SPRUNG
13683
13684  051676  010437  000004                  MOV     R4,@#4                  ;RESTORE BUS ERROR VECTOR
13685  051702  005004                          CLR     R4                      ;[R4] = S / B SP
13686  051704  010603                          MOV     SP,R3                   ;[R3] = WAS SP
13687  051706  010506                          MOV     R5,SP                   ;RESET THE SP
13688  051710  104003          3$:     ERROR   3                               ;TRAP NOT SPRUNG
13689  051712  000414                          BR      TST707                  ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13690
13691  051714  022706  000000          4$:     CMP     #0,SP                   ;WAS IT A RED ZONE TRAP ?
13692  051720  001406                          BEQ     6$                      ;BR IF YES
```

```
13693
13694  051722  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13695  051726  005004                  CLR     R4              ;[R4]= S / 0 SP
13696  051730  010403                  MOV     SP,R3           ;[R3] = WAS SP
13697  051732  010506                  MOV     R5,SP           ;RESET THE SP
13698  051734  104003          5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13699
13700  051736  010506          6$:     MOV     R5,SP           ;FIX UP THE SP
13701  051740  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13702
13703                          ;;*****************************************************************
13704                          ;*TEST 707      RSVD INSTRUCTION TEST - 000007 THRU 000077
13705                          ;;*****************************************************************
13706  051744                  ^ST707:
13707  051744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13708  051746  012700  000707          MOV     #707,R0         ;;LOAD R0 WITH TEST NUMBER
13709  051752  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13710  051754  012737  052012  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13711  051762  005037  000012          CLR     @#12
13712  051766  012701  000007          MOV     #7,R1           ;SET UP FIRST ONE IN GROUP
13713  051772  010737  001010          MOV     PC,@#$LPERR     ;ONLY LOOP ON BAD OP CODE
13714  051776  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13715  052000  010137  052006          MOV     R1,@#2$ ;LOAD NEW INSTR
13716  052004  000257                  CCC                     ;SCOPE SYNC
13717
13718  052006  000007          2$:     000007                  ;TEST THE RSVD INSTR - THIS LOCATION
13719                                                          ;GETS CHANGED EACH PASS THROUGH
13720
13721  052010  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13722
13723  052012  005201          4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13724  052014  022701  000100          CMP     #100,R1         ;AT END OF THIS GROUP ??
13725  052020  001366                  BNE     1$              ;BR IF NOT
13726
13727  052022  010506                  MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13728  052024  012737  051752  001010  MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13729                          ;;*****************************************************************
13730                          ;*TEST 710      RSVD INSTRUCTION TEST - 000210 THRU 000237
13731                          ;;*****************************************************************
13732  052032                  ^ST710:
13733  052032  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13734  052034  012700  000710          MOV     #710,R0         ;;LOAD R0 WITH TEST NUMBER
13735  052040  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13736  052042  012737  052100  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13737  052050  005037  000012          CLR     @#12
13738  052054  012701  000210          MOV     #210,R1         ;SET UP FIRST ONE IN GROUP
13739  052060  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13740  052064  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13741  052066  010137  052074          MOV     R1,@#2$ ;LOAD NEW INSTR
13742  052072  000257                  CCC                     ;SCOPE SYNC
13743
13744  052074  000210          2$:     000210                  ;TEST THE RSVD INSTR - THIS LOCATION
13745                                                          ;GETS CHANGED EACH PASS THROUGH
13746
13747  052076  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13748
```

```
13749  052100  005201                        4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13750  052102  022701  000240                        CMP     #240,R1         ;AT END OF THIS GROUP ??
13751  052106  001366                              BNE     1$              ;BR IF NOT
13752
13753  052110  010506                              MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13754  052112  012737  052040  001010              MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13755
13756                                  ;;*****************************************************************
13757                                  ;*TEST 711      RSVD INSTRUCTION TEST - 007000 THRU 007777
13758                                  ;;*****************************************************************
13759  052120                          TST711:
13760  052120  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13761  052122  012700  000711                      MOV     #711,R0         ;;LOAD R0 WITH TEST NUMBER
13762  052126  010605                        5$:   MOV     SP,R5           ;SAVE THE SP
13763  052130  012737  052166  000010              MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13764  052136  005037  000012                      CLR     @#12
13765  052142  012701  007000                      MOV     #7000,R1        ;SET UP FIRST ONE IN GROUP
13766  052146  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13767  052152  010506                        1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13768  052154  010137  052162                      MOV     R1,@#2$ ;LOAD NEW INSTR
13769  052160  000257                              CCC                     ;SCOPE SYNC
13770
13771  052162  007000                        2$:   007000                  ;TEST THE RSVD INSTR - THIS LOCATION
13772                                                                       ;GETS CHANGED EACH PASS THROUGH
13773
13774  052164  104005                        3$:   ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13775
13776  052166  005201                        4$:   INC     R1              ;GENERATE NEW RSVD INSTR
13777  052170  022701  010000                      CMP     #10000,R1       ;AT END OF THIS GROUP ??
13778  052174  001366                              BNE     1$              ;BR IF NOT
13779
13780  052176  010506                              MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13781  052200  012737  052126  001010              MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13782
13783                                  ;;*****************************************************************
13784                                  ;*TEST 712      RSVD INSTRUCTION TEST - 075000 THRU 076777
13785                                  ;;*****************************************************************
13786  052206                          TST712:
13787  052206  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13788  052210  012700  000712                      MOV     #712,R0         ;;LOAD R0 WITH TEST NUMBER
13789  052214  010605                        5$:   MOV     SP,R5           ;SAVE THE SP
13790  052216  012737  052256  000010              MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13791  052224  005037  000012                      CLR     @#12
13792  052230  012701  075000                      MOV     #75000,R1       ;SET UP FIRST ONE IN GROUP
13793  052234  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13794  052240  010506                        1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13795  052242  010137  052250                      MOV     R1,@#2$ ;LOAD NEW INSTR
13796  052246  000257                              CCC                     ;SCOPE SYNC
13797
13798  052250  075000                        2$:   75000                   ;TEST THE RSVD INSTR - THIS LOCATION
13799                                                                       ;GETS CHANGED EACH PASS THROUGH
13800
13801  052252  000240                              NOP                     ;IN CASE NON TRAPPING INSTR IS TWO WORDS
13802  052254  104005                        3$:   ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13803
13804  052256  005201                        4$:   INC     R1              ;GENERATE NEW RSVD INSTR
```

```
MAINDEC-11-DQKDA-B   KD11-K BASIC LOGIC TESTS   MACY11 27(1006)  25-APR-77  08:37  PAGE 250
DQKDAB.P11    25-APR-77 08:29                   T712    RSVD INSTRUCTION TEST - 075000 THRU 076777

 13805  052260  022701  076600                         CMP     #MED,R1         ;MED INSTRUCTION?
 13806  052264  001774                                 BEQ     4$              ;BR IF YES--SKIP IT.
 13807  052266  022701  077000                         CMP     #077000,R1      ;AT END OF THIS GROUP ??
 13808  052272  001362                                 BNE     1$              ;BR IF NOT
 13809
 13810  052274  010506                                 MOV     R5,SP           ;MAKE SURE TO RESET THE SP
 13811  052276  012737  052214  001010                 MOV     #5$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
 13812
 13813                                  ;;*****************************************************************
 13814                                  ;*TEST 713      RSVD INSTRUCTION TEST - 106400 THRU 107777
 13815                                  ;;*****************************************************************
 13816  052304                          TST713:
 13817  052304  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 13818  052306  012700  000713                 MOV     #713,R0         ;;LOAD R0 WITH TEST NUMBER
 13819  052312  010605                  5$:    MOV     SP,R5           ;SAVE THE SP
 13820  052314  012737  052352  000010         MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
 13821  052322  005037  000012                 CLR     @#12
 13822  052326  012701  106400                 MOV     #106400,R1      ;SET UP FIRST ONE IN GROUP
 13823  052332  010737  001010                 MOV     PC,@#SLPERR     ;SET ERROR LOOP ADDRESS
 13824  052336  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
 13825  052340  010137  052346                 MOV     R1,@#2$ ;LOAD NEW INSTR
 13826  052344  000257                         CCC                     ;SCOPE SYNC
 13827
 13828  052346  106400                  2$:    106400                  ;TEST THE RSVD INSTR - THIS LOCATION
 13829                                                                  ;GETS CHANGED EACH PASS THROUGH
 13830
 13831  052350  104005                  3$:    ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
 13832
 13833  052352  005201                  4$:    INC     R1              ;GENERATE NEW RSVD INSTR
 13834  052354  022701  106500                 CMP     #106500,R1      ;MFPD INSTRUCTION ??
 13835  052360  001002                         BNE     10$             ;BR IF NOT
 13836  052362  012701  106700                 MOV     #106700,R1      ;SKIP MFPD AND MTPD INSTRUCTIONS
 13837  052366  022701  110000          10$:   CMP     #110000,R1      ;AT END OF THIS GROUP ??
 13838  052372  001361                         BNE     1$              ;BR IF NOT
 13839
 13840  052374  010506                         MOV     R5,SP           ;MAKE SURE TO RESET THE SP
 13841  052376  012737  052312  001010         MOV     #5$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
 13842  052404  012737  061122  000010         MOV     #RSERR,@#10     ;RESTORE RSVD INSTR VECTOR
 13843  052412  012737  000340  000012         MOV     #340,@#12
 13844  052420  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 13845
 13846                                  ;THIS NEXT GROUP OF SEQUENTIAL TESTS VERIFIES THAT A "T" BIT
 13847                                  ;TRAP CAN BE SERVICED IN EACH MICROWORD THAT DOES A "BUT SERVICE"
 13848                                  ;EACH ROUTINE ENTERS THE TRAP MICROUTINE WHEN THE TRAP IS SPRUNG
 13849
 13850  052422  012737  061070  000014  TSET:  MOV     #TBSER,@#14     ;SET UP THE "T" BIT TRAP VECTOR
 13851  052430  012737  000340  000016         MOV     #340,@#16       ;PRIORITY 7
 13852
 13853                                  ;;*****************************************************************
 13854                                  ;*TEST 714      BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
 13855                                  ;"INSTAB" (INSTRUCTION TABLE) CONTAINS ALL ONE WORD INSTRUCTIONS
 13856                                  ;THAT TEST A "BUT SERVICE" IN A UNIQUE ROM LOCATION.  THE TABLE MUST
 13857                                  ;BE TERMINATED WITH A 0 ENTRY.
 13858                                  ;;*****************************************************************
 13859  052436                          TST714:
 13860  052436  012700  000714                 MOV     #714,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
13861  052442  010605                  6$:     MOV     SP,R5           ;SAVE THE SP
13862  052444  012704  063636                  MOV     #INSTAB,R4      ;PUT POINTER TO TABLE IN R4
13863  052450  012401                  4$:     MOV     (R4)+,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13864  052452  001422                          BEQ     5$              ;EXIT TEST IF END OF TABLE
13865  052454  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ON FAILING INSTRUCTION ONLY
13866  052460  010137  052512          1$:     MOV     R1,@#2$         ;STORE TEST INSTRUCTION TO BE EXECUTED
13867  052464  012702  063312                  MOV     #^^F0,R2        ;IN CASE DM1 DEST--(R2)
13868  052470  012703  063316                  MOV     #^^F1,R3        ;IN CASE SM1--(R3)
13869  052474  010506                          MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13870  052476  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13871  052502  012746  052512                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13872  052506  000257                          CCC                     ;SCOPE SYNC
13873  052510  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13874
13875  052512  000240                  2$:     NOP                     ;INSTRUCTION FROM TABLE IS STORED HERE AND
13876                                                                  ;SHOULD SPRING TRAP
13877
13878  052514  104005                  3$:     ERROR   5               ;BUT SERVICE FAILED
13879
13880  052516  000754                          BR      4$              ;GET NEXT INSTRUCTION FOR BUT SERVICE TEST
13881  052520  012737  052442  001010  5$:     MOV     #6$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13882
13883                                  ;;*****************************************************************
13884                                  ;*TEST 715      BUT SERVICE TEST - (RTI)
13885                                  ;;*****************************************************************
13886  052526                          TST715:
13887  052526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13888  052530  012700  000715                  MOV     #715,R0         ;;LOAD R0 WITH TEST NUMBER
13889  052534  013701  052552                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13890  052540  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13891  052544  012746  052554                  MOV     #3$,-(SP)       ;MAKE NEW PC = 3$
13892  052550  000257                          CCC                     ;SCOPE SYNC
13893
13894  052552  000002                  2$:     RTI                     ;INSTRUCTION SHOULD SPRING TRAP
13895
13896  052554  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13897
13898                                  ;;*****************************************************************
13899                                  ;*TEST 716      BUT SERVICE TEST - (JSR  %R,@A)
13900                                  ;;*****************************************************************
13901  052556                          TST716:
13902  052556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13903  052560  012700  000716                  MOV     #716,R0         ;;LOAD R0 WITH TEST NUMBER
13904  052564  013701  052634                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13905                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT14
13906  052570  032737  040000  063234          BIT     #BIT14,@#BPTLOC ;BREAKPOINT HALT SET ??
13907  052576  001401                          BEQ     .+4             ;BR IF NOT
13908  052600  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
13909  052602  010605                          MOV     SP,R5           ;SAVE THE SP
13910  052604  010737  001010                  MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13911  052610  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13912  052612  012737  052640  063316          MOV     #3$,@#MBUF1     ;SET UP POINTER--DEST ADDR = 3$ FOR JSR
13913  052620  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13914  052624  012746  052634                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13915  052630  000257                          CCC                     ;SCOPE SYNC
13916  052632  000006                          RTT                     ;SET "T" BIT - GO TO 2$
```

```
13917
13918  052634  004777  010456          2$:     JSR     PC,@MBUF1       ;INSTRUCTION SHOULD SPRING TRAP
13919
13920  052640  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13921
13922  052642  010506                          MOV     R5,SP           ;RESTORE SP IF ALL OK OR NOT LOOPING
13923                                  ;;*******************************************************************
13924                                  ;*TEST 717       BUT SERVICE TEST - (JMP A)
13925                                  ;;*******************************************************************
13926  052644                          TST717:
13927  052644  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13928  052646  012700  000717                  MOV     #717,R0         ;;LOAD R0 WITH TEST NUMBER
13929  052652  013701  052672                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13930  052656  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13931  052662  012746  052672                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13932  052666  000257                          CCC                     ;SCOPE SYNC
13933  052670  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13934
13935  052672  000167  000000          2$:     JMP     3$              ;JMP INSTRUCTION SHOULD SPRING TRAP
13936
13937  052676  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13938
13939                                  ;;*******************************************************************
13940                                  ;*TEST 720       BUT SERVICE TEST - (JMP @A)
13941                                  ;;*******************************************************************
13942  052700                          TST720:
13943  052700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13944  052702  012700  000720                  MOV     #720,R0         ;;LOAD R0 WITH TEST NUMBER
13945  052706  013701  052734                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13946  052712  012737  052740  063312          MOV     #3$,@#MBUF0     ;SET UP POINTER--DEST ADDR = 3$ FOR JMP
13947  052720  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13948  052724  012746  052734                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13949  052730  000257                          CCC                     ;SCOPE SYNC
13950  052732  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13951
13952  052734  000177  010352          2$:     JMP     @MBUF0          ;JMP INSTRUCTION SHOULD SPRING TRAP
13953
13954  052740  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13955
13956                                  ;;*******************************************************************
13957                                  ;*TEST 721       BUT SERVICE TEST - (RTS PC)
13958                                  ;;*******************************************************************
13959  052742                          TST721:
13960  052742  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13961  052744  012700  000721                  MOV     #721,R0         ;;LOAD R0 WITH TEST NUMBER
13962  052750  013701  053004                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13963  052754  010605                          MOV     SP,R5           ;SAVE THE SP
13964  052756  010737  001010                  MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13965  052762  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13966  052764  012746  053006                  MOV     #3$,-(SP)       ;RTS WILL LOAD PC WITH 3$
13967  052770  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13968  052774  012746  053004                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13969  053000  000257                          CCC                     ;SCOPE SYNC
13970  053002  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13971
13972  053004  000207          2$:     RTS     PC              ;RTS INSTRUCTION SHOULD SPRING TRAP
```

```
13973
13974  053006  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13975
13976                                  ;;*****************************************************************
13977                                  ;TEST 722       ALU ADD FUNCTION TEST
13978                                  ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13979                                  ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13980                                  ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13981
13982                                  ;       AIN     BIN     CIN
13983
13984                                  ;       0       0       0
13985                                  ;       0       0       1
13986                                  ;       0       1       0
13987                                  ;       0       1       1
13988                                  ;       1       0       0
13989                                  ;       1       0       1
13990                                  ;       1       1       0
13991                                  ;       1       1       1
13992
13993                                  ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
13994                                  ;TAGGED "ALUADD" AS SHOWN BELOW:
13995
13996                                          ;ALUADD:        NULL
13997                                  ;                       SRC OP1
13998                                  ;                       DST OP1
13999                                  ;                       SUM1
14000                                  ;                       SRC OP2
14001                                  ;                       DST OP2
14002                                  ;                       SUM2
14003                                  ;                       ETC.
14004
14005                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14006                                  ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14007                                  ;;*****************************************************************
14008  053010                          TST722:
14009  053010  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14010  053012  012700  000722                  MOV     #722,R0                 ;;LOAD R0 WITH TEST NUMBER
14011  053016  012705  063340          1$:     MOV     #ALUADD+4,R5            ;R5 POINTS TO TABLE OF NO.S
14012  053022  010737  001010                  MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
14013  053026  024545                          CMP     -(R5),-(R5)             ;RESET R5 TO POINT TO BAD GUYS
14014                                                                           ;(OR NULL ENTRY FIRST TIME THROUGH)
14015  053030  005725                  4$:     TST     (R5)+                   ;POINT TO A SRC OP
14016  053032  022705  063416                  CMP     #ALUADD+62,R5           ;DONE ALL NO.S IN TABLE ?
14017  053036  001413                          BEQ     5$                      ;BR IF YES
14018  053040  012501                          MOV     (R5)+,R1                ;LOAD SRC OP
14019  053042  012503                          MOV     (R5)+,R3                ;LOAD DEST OP
14020  053044  000257                          CCC                             ;SCOPE SYNC
14021
14022  053046  060103                  2$:     ADD     R1,R3                   ;TEST THE ADD FUNCTION
14023
14024  053050  021503                          CMP     (R5),R3                 ;CORRECT SUM ?
14025  053052  001766                          BEQ     4$                      ;GO ADD NEXT PAIR IF YES
14026
14027  053054  011504                          MOV     (R5),R4                 ;GET S / B SUM
14028  053056  014502                          MOV     -(R5),R2                ;GET DEST OP
```

```
14029  053060  104010                  3$:     ERROR   10              ;ALU ADD OPERATION FAILED
14030
14031  053062  005725                          TST     (R5)+           ;CORRECT R5 POINTER
14032  053064  000761                          BR      4$              ;GO DO NEXT PAIR
14033
14034  053066  012737  053016  001010  5$:     MOV     #1$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
14035
14036                                  ;;*****************************************************************
14037                                  ;*TEST 723      ALU SUB FUNCTION TEST
14038                                  ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
14039                                  ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
14040                                  ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
14041
14042                                  ;       AIN     BIN     CIN
14043
14044                                  ;       0       0       0
14045                                  ;       0       0       1
14046                                  ;       0       1       0
14047                                  ;       0       1       1
14048                                  ;       1       0       0
14049                                  ;       1       0       1
14050                                  ;       1       1       0
14051                                  ;       1       1       1
14052
14053                                  ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
14054                                  ;TAGGED "ALUADD" AS SHOWN BELOW:
14055
14056                                          ;ALUSUB:        NULL
14057                                  ;                       SRC OP1
14058                                  ;                       DST OP1
14059                                  ;                       DIFF1
14060                                  ;                       SRC OP2
14061                                  ;                       DST OP2
14062                                  ;                       DIFF2
14063                                  ;                       ETC.
14064
14065                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14066                                  ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14067                                  ;;*****************************************************************
14068  053074                          TST723:
14069  053074  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14070  053076  012700  000723                  MOV     #723,R0         ;;LOAD R0 WITH TEST NUMBER
14071  053102  012705  063560          1$:     MOV     #ALUSUB+4,R5    ;R5 POINTS TO TABLE OF NO.S
14072  053106  010737  001010                  MOV     PC,@#SLPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14073  053112  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14074                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
14075  053114  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
14076  053116  022705  063636                  CMP     #ALUSUB+62,R5   ;DONE ALL NO.S IN TABLE ?
14077  053122  001413                          BEQ     5$              ;BR IF YES
14078  053124  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
14079  053126  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
14080  053130  000257                          CCC                     ;SCOPE SYNC
14081
14082  053132  160103                  2$:     SUB     R1,R3           ;TEST THE SUB FUNCTION
14083
14084  053134  021503                          CMP     (R5),R3         ;CORRECT DIFF. ?
```

```
14085  053136  001766                        BEQ     4$              ;GO SUB NEXT PAIR IF YES
14086
14087  053140  011504                        MOV     (R5),R4         ;GET S / B DIFF
14088  053142  014502                        MOV     -(R5),R2        ;GET DEST OP
14089  053144  104010                3$:     ERROR   10              ;ALU SUB OPERATION FAILED
14090
14091  053146  005725                        TST     (R5)+           ;CORRECT R5 POINTER
14092  053150  000761                        BR      4$              ;GO DO NEXT PAIR
14093
14094  053152  012737  053102  001010  5$:     MOV     #1$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
14095
14096                                  ;;*****************************************************************
14097                                  ;TEST 724       ALU "AND" FUNCTION TEST USING BIC INSTRUCTION
14098                                  ;THIS TEST VERIFIES THAT THE ALU "AND" FUNCTION RESPONDS CORRECTLY
14099                                  ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
14100                                  ;IT  EXECUTES THE BIC INSTRUCTION FOR THE FOLLOWING PAIRS OF
14101                                  ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
14102
14103                                          ;SOURCE OP      DEST. OP        RESULT
14104
14105                                          ;000000         000000          000000
14106                                          ;177777         177777          000000
14107                                          ;000000         177777          177777
14108                                          ;177777         000000          000000
14109                                          ;125252         12[illegible]2          000000
14110                                          ;052525         05[illegible]5          000000
14111                                          ;125252         05[illegible]5          052525
14112                                          ;052525         12[illegible]2          125252
14113
14114                                  ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TEBLE TAGGED
14115                                  ;"ANDTAB" IN THE FOLLOWING PATTERN:
14116                                          ;ANDTAB:        NULL
14117                                  ;                       SRC OP1
14118                                  ;                       DST OP1
14119                                  ;                       ANS1
14120                                  ;                       SRC OP2
14121                                  ;                       DST OP2
14122                                  ;                       ANS2
14123                                  ;                       ETC.
14124
14125                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
14126                                  ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
14127                                  ;;*****************************************************************
14128  053160                          TST724:
14129  053160  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14130  053162  012700  000724                MOV     #724,R0         ;;LOAD R0 WITH TEST NUMBER
14131  053166  012705  063420        1$:     MOV     #ANDTAB+4,R5    ;R5 POINTS TO TABLE OF TEST NO.S
14132  053172  010737  001010                MOV     PC,@#SLPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14133  053176  024545                        CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14134                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
14135  053200  005725                4$:     TST     (R5)+           ;POINT TO A SOURCE OPR
14136  053202  022705  063476                CMP     #ANDTAB+62,R5   ;DONE ALL COMBINATIONS ?
14137  053206  001413                        BEQ     5$              ;BR IF YES
14138  053210  012501                        MOV     (R5)+,R1        ;LOAD THE SRC OP
14139  053212  012503                        MOV     (R5)+,R3        ;LOAD THE DEST OP
14140  053214  000257                        CCC                     ;SCOPE SYNC
```

```
14141
14142  053216  040103                  25:     BIC     R1,R3           ;TEST THE "AND"
14143
14144  053220  020315                          CMP     R3,(R5)         ;RESULT CORRECT ?
14145  053222  001766                          BEQ     4$              ;BR IF YES - GET THE NEXT PAIR
14146
14147  053224  011504                          MOV     (R5),R4         ;GET THE S / B DATA
14148  053226  014502                          MOV     -(R5),R2        ;GET DEST OP
14149  053230  104010                  3$:     ERROR   10              ;ALU "AND" FAILED
14150
14151  053232  005725                          TST     (R5)+           ;CORRECT R5 POINTER
14152  053234  000761                          BR      4$              ;GO GET NEXT PAIR
14153
14154  053236  012737  053166  001010  5$:     MOV     #1$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
14155
14156                                  ;;*****************************************************************
14157                                  ;*TEST 725       ALU "OR" FUNCTION TEST USING BIS INSTRUCTION
14158                                  ;THIS TEST VERIFIES THAT THE ALU "OR" FUNCTION RESPONDS CORRECTLY
14159                                  ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
14160                                  ;IT  EXECUTES THE BIS INSTRUCTION FOR THE FOLLOWING PAIRS OF
14161                                  ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
14162
14163                                          ;SOURCE OP      DEST. OP        RESULT
14164
14165                                          ;000000         000000          000000
14166                                          ;177777         177777          177777
14167                                          ;000000         177777          177777
14168                                          ;177777         000000          177777
14169                                          ;125252         125252          125252
14170                                          ;052525         052525          052525
14171                                          ;125252         052525          177777
14172                                          ;052525         125252          177777
14173
14174                                  ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TABLE TAGGED
14175                                  ;"ORTAB" IN THE FOLLOWING PATTERN:
14176                                          ;ORTAB: NULL
14177                                  ;                       SRC OP1
14178                                  ;                       DST OP1
14179                                  ;                       ANS1
14180                                  ;                       SRC OP2
14181                                  ;                       DST OP2
14182                                  ;                       ANS2
14183                                  ;                       ETC.
14184
14185                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
14186                                  ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
14187                                  ;;*****************************************************************
14188  053244                          TST725:
14189  053244  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14190  053246  012700  000725                  MOV     #725,R0         ;LOAD R0 WITH TEST NUMBER
14191  053252  012705  063500          1$:     MOV     #ORTAB+4,R5     ;R5 POINTS TO TABLE OF TEST NO.S
14192  053256  010737  001010                  MOV     PC,@#SLPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14193  053262  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14194                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
14195  053264  005725                  4$:     TST     (R5)+           ;POINT TO A SOURCE OPR
14196  053266  022705  063556                  CMP     #ORTAB+62,R5    ;DONE ALL COMBINATIONS ?
```

```
14197  053272  001413                        BEQ     5$              ;BR IF YES
14198  053274  012501                        MOV     (R5)+,R1        ;LOAD THE SRC OP
14199  053276  012503                        MOV     (R5)+,R3        ;LOAD THE DEST OP
14200  053300  000257                        CCC                     ;SCOPE SYNC
14201
14202  053302  050103                 2$:    BIS     R1,R3           ;TEST THE "OR"
14203
14204  053304  020315                        CMP     R3,(R5)         ;RESULT CORRECT ?
14205  053306  001766                        BEQ     4$              ;BR IF YES - GET THE NEXT PAIR
14206
14207  053310  011504                        MOV     (R5),R4         ;GET THE S / B DATA
14208  053312  014502                        MOV     -(R5),R2        ;GET DEST OP
14209  053314  104010                 3$:    ERROR   10              ;ALU "OR" FAILED
14210
14211  053316  005725                        TST     (R5)+           ;CORRECT R5 POINTER
14212  053320  000761                        BR      4$              ;GO GET NEXT PAIR
14213
14214  053322  012737  053252  001010  5$:    MOV     #1$,@#SLPERR    ;LOOP FROM BEGINNING ON ERROR
14215
14216                                 ;;*****************************************************************
14217                                 ;*TEST 726      INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
14218                                 ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
14219                                 ;TEST SEQUENCE:
14220
14221                                         ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14222                                         ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14223                                         ;3. THE SOURCE OP IS INCREMENTED
14224                                         ;4. THE DEST OP IS DECREMENTED
14225                                         ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE SOURCE OP GOES
14226                                         ;   NEGATIVE
14227
14228                                 ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14229
14230                                 ;       1. SW09=0  THE TEST IS EXITED
14231                                 ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14232
14233                                 ;;*****************************************************************
14234  053330                         TST726:
14235  053330  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14236  053332  012700  000726                MOV     #726,R0         ;;LOAD R0 WITH TEST NUMBER
14237  053336  005001                 10$:   CLR     R1              ;INITIALIZE REGS TO 000000
14238  053340  005002                        CLR     R2
14239  053342  005004                        CLR     R4
14240  053344  010737  001010                MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14241  053350  010203                 1$:    MOV     R2,R3           ;LOAD DEST OPERAND
14242  053352  000257                        CCC                     ;SCOPE SYNC
14243
14244  053354  060103                 2$:    ADD     R1,R3           ;ADD THE TWO TEST NO.S
14245                                                                 ;RESULT S / B = 000000
14246
14247  053356  020403                        CMP     R4,R3           ;RESULT = 000000 ?
14248  053360  001402                        BEQ     4$              ;BR IF YES
14249
14250  053362  104010                 3$:    ERROR   10              ;INCORRECT RESULT IN R3
14251
14252  053364  000407                        BR      TST727          ;;EXIT TO NEXT TEST
```

```
14253
14254  053366  005201                  4$:     INC     R1              ;ADD 1 TO SOURCE OP
14255  053370  100402                          BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14256  053372  005302                          DEC     R2              ;SUB 1 FROM THE DEST OP
14257  053374  000765                          BR      1$              ;GO ADD THE TWO NO.S
14258
14259  053376  012737  053336  001010  5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14260
14261                                  ;;*****************************************************************
14262                                  ;#TEST 727      INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
14263                                  ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
14264                                  ;TEST SEQUENCE:
14265
14266                                          ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14267                                          ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14268                                          ;3. THE SOURCE OP IS DECREMENTED
14269                                          ;4. THE DEST OP IS INCREMENTED
14270                                          ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE DEST. OP GOES
14271                                          ;   NEGATIVE
14272
14273                                  ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14274
14275                                  ;       1. SW09=0  THE TEST IS EXITED
14276                                  ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14277                                  ;;*****************************************************************
14278  053404                          TST727:
14279  053404  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14280  053406  012700  000727                  MOV     #727,R0         ;;LOAD R0 WITH TEST NUMBER
14281  053412  005001                  10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14282  053414  005002                          CLR     R2
14283  053416  005004                          CLR     R4
14284  053420  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14285  053424  010203                  1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14286  053426  000257                          CCC                     ;SCOPE SYNC
14287
14288  053430  060103                  2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14289                                                                  ;RESULT S / B = 000000
14290
14291  053432  020403                          CMP     R4,R3           ;RESULT = 000000 ?
14292  053434  001402                          BEQ     4$              ;BR IF YES
14293
14294  053436  104010                  3$:     ERROR   10              ;INCORRECT RESULT IN R3
14295
14296  053440  000407                          BR      TST730          ;;GO TO SCOPE EXIT
14297
14298  053442  005202                  4$:     INC     R2              ;ADD 1 TO DEST. OP
14299  053444  100402                          BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14300  053446  005301                          DEC     R1              ;SUB 1 FROM THE SOURCE OP
14301  053450  000765                          BR      1$              ;GO ADD THE TWO NO.S
14302
14303  053452  012737  053412  001010  5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14304
14305                                  ;;*****************************************************************
14306                                  ;#TEST 730      MUL RA,RB TEST ; N:C = 1111
14307                                  ;;*****************************************************************
14308  053460                          TST730:
```

```
14309  053460  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14310  053462  012700  000730                  MOV     #730,R0         ;;LOAD R0 WITH TEST NUMBER
14311  053466  013737  053516  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14312  053474  005001                          CLR     R1              ;S/B RESULT IN R2
14313  053476  012704  000006                  MOV     #6,R4           ;S/B RESULT IN R3
14314  053502  012702  001002                  MOV     #2,R2           ;INITIALIZE REG
14315  053506  005003                          CLR     R3              ;INITIALIZE REG + 1
14316  053510  012705  000003                  MOV     #3,R5           ;INITIALIZE SRC
14317  053514  000277                          SCC                     ;SCOPE SYNC
14318
14319  053516  070205                  2$:     MUL     R5,R2           ;TEST THE MUL
14320
14321  053520  100403                          BMI     3$              ;N:C=0000?
14322  053522  001402                          BEQ     3$
14323  053524  102401                          BVS     3$
14324  053526  103001                          BCC     4$
14325
14326  053530  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14327
14328  053532  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14329  053534  001002                          BNE     5$              ;BR IF NOT
14330  053536  020102                          CMP     R1,R2           ;REG CORRECT?
14331  053540  001401                          BEQ     TST731          ;;BR IF YES
14332
14333  053542  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14334
14335                                  ;;*****************************************************************
14336                                  ;*TEST 731      MUL (RA),RB TEST ; N:C = 0000-SET C
14337                                  ;;*****************************************************************
14338  053544                          TST731:
14339  053544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14340  053546  012700  000731                  MOV     #731,R0         ;;LOAD R0 WITH TEST NUMBER
14341  053552  013737  053606  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14342  053560  005001                          CLR     R1              ;S/B RESULT IN R2
14343  053562  012704  123450                  MOV     #123450,R4      ;S/B RESULT IN R3
14344  053566  012702  012345                  MOV     #012345,R2      ;INITIALIZE REG
14345  053572  005003                          CLR     R3              ;INITIALIZE REG + 1
14346  053574  012705  063312                  MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14347  053600  012715  000010                  MOV     #10,(R5)        ;INITIALIZE SRC
14348  053604  000257                          CCC                     ;SCOPE SYNC
14349
14350  053606  070215                  2$:     MUL     (R5),R2         ;TEST THE MUL
14351
14352  053610  100403                          BMI     3$              ;N:C=0001?
14353  053612  001402                          BEQ     3$
14354  053614  102401                          BVS     3$
14355  053616  103401                          BCS     4$
14356
14357  053620  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14358
14359  053622  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14360  053624  001002                          BNE     5$              ;BR IF NOT
14361  053626  020102                          CMP     R1,R2           ;REG CORRECT?
14362  053630  001401                          BEQ     TST732          ;;BR IF YES
14363
14364  053632  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
```

```
14365
14366                                   ;;*****************************************************************
14367                                   ;*TEST 732      MUL (RA)+,RB TEST ; N:C = 0000-SET Z
14368                                   ;;*****************************************************************
14369  053634                           TST732:
14370  053634  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14371  053636  012700  000732                   MOV     #732,R0         ;;LOAD R0 WITH TEST NUMBER
14372  053642  013737  053674  001076           MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14373  053650  005001                           CLR     R1              ;S/B RESULT IN R2
14374  053652  005004                           CLR     R4              ;S/B RESULT IN R3
14375  053654  005002                           CLR     R2              ;INITIALIZE REG
14376  053656  012703  177777                   MOV     #-1,R3          ;INITIALIZE REG + 1
14377  053662  012705  063312                   MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14378  053666  012715  000010                   MOV     #10,(R5)        ;INITIALIZE SRC
14379  053672  000257                           CCC                     ;SCOPE SYNC
14380
14381  053674  070225                   2$:     MUL     (R5)+,R2        ;TEST THE MUL
14382
14383  053676  100403                           BMI     3$              ;N:C=0100?
14384  053700  001002                           BNE     3$
14385  053702  102401                           BVS     3$
14386  053704  103001                           BCC     4$
14387
14388  053706  104044                   3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14389
14390  053710  020304                   4$:     CMP     R3,R4           ;REG+1 CORRECT?
14391  053712  001002                           BNE     5$              ;BR IF NOT
14392  053714  020102                           CMP     R1,R2           ;REG CORRECT?
14393  053716  001401                           BEQ     6$              ;BR IF YES
14394
14395  053720  104045                   5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14396
14397  053722  022705  063314           6$:     CMP     #MBUF0+2,R5     ;DID R5 GET AUTO-INCREMENTED?
14398  053726  001401                           BEQ     TST733          ;;BR IF YES
14399
14400  053730  104046                           ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14401
14402                                   ;;*****************************************************************
14403                                   ;*TEST 733      MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST = -,+
14404                                   ;;*****************************************************************
14405  053732                           TST733:
14406  053732  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14407  053734  012700  000733                   MOV     #733,R0         ;;LOAD R0 WITH TEST NUMBER
14408  053740  013737  054000  001076           MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14409  053746  012701  177777                   MOV     #-1,R1          ;S/B RESULT IN R2
14410  053752  012704  177770                   MOV     #-10,R4         ;S/B RESULT IN R3
14411  053756  012702  000001                   MOV     #1,R2           ;INITIALIZE REG
14412  053762  005003                           CLR     R3              ;INITIALIZE REG + 1
14413  053764  012705  063306                   MOV     #ATA+10,R5      ;SET UP POINTER TO POINTER TO MBUF0
14414  053770  012737  177770  063312           MOV     #-10,@#MBUF0    ;INITIALIZE SRC
14415  053776  000257                           CCC                     ;SCOPE SYNC
14416
14417  054000  070235                   2$:     MUL     @(R5)+,R2       ;TEST THE MUL
14418
14419  054002  100003                           BPL     3$              ;N:C=1000?
14420  054004  001402                           BEQ     3$
```

```
14421  054006  102401                        BVS     3$
14422  054010  103001                        BCC     4$
14423
14424  054012  104044              3$:       ERROR   44              ;COND CODES SET IMPROPERLY
14425
14426  054014  020304              4$:       CMP     R3,R4           ;REG+1 CORRECT?
14427  054016  001002                        BNE     5$              ;BR IF NOT
14428  054020  020102                        CMP     R1,R2           ;REG CORRECT?
14429  054022  001401                        BEQ     6$              ;BR IF YES
14430
14431  054024  104045              5$:       ERROR   45              ;MUL DELIVERED WRONG RESULT
14432
14433  054026  022705  063310      6$:       CMP     #ATA+12,R5      ;DID R5 GET AUTO-INCREMENTED?
14434  054032  001401                        BEQ     TST734          ;;BR IF YES
14435
14436  054034  104046                        ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14437
14438                          ;;**********************************************************************
14439                          ;*TEST 734      MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK = +,-
14440                          ;;**********************************************************************
14441  054036                  TST734:
14442  054036  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14443  054040  012700  000734               MOV     #734,R0         ;;LOAD R0 WITH TEST NUMBER
14444  054044  013737  054104  001076       MOV     @#2$,@#TMP0     ;GET TEST INSTRUCTION WORD
14445  054052  012701  177777               MOV     #-1,R1          ;S/B RESULT IN R2
14446  054056  012704  177770               MOV     #-10,R4         ;S/B RESULT IN R3
14447  054062  012702  177777               MOV     #-1,R2          ;INITIALIZE REG
14448  054066  005003                        CLR     R3              ;INITIALIZE REG + 1
14449  054070  012705  063314               MOV     #MBUF0+2,R5     ;SET UP POINTER TO SRC
14450  054074  012737  000010  063312       MOV     #10,@#MBUF0     ;INITIALIZE SRC
14451  054102  000277                        SCC                     ;SCOPE SYNC
14452
14453  054104  070245              2$:       MUL     -(R5),R2        ;TEST THE MUL
14454
14455  054106  100003                        BPL     3$              ;N:C=1000?
14456  054110  001                           BEQ     3$
14457  054112  102                           BVS     3$
14458  054114  103001                        BCC     4$
14459
14460  054116  104044              3$:       ERROR   44              ;COND CODES SET IMPROPERLY
14461
14462  054120  020304              4$:       CMP     R3,R4           ;REG+1 CORRECT?
14463  054122  001002                        BNE     5$              ;BR IF NOT
14464  054124  020102                        CMP     R1,R2           ;REG CORRECT?
14465  054126  001401                        BEQ     6$              ;BR IF YES
14466
14467  054130  104045              5$:       ERROR   45              ;MUL DELIVERED WRONG RESULT
14468
14469  054132  022705  063312      6$:       CMP     #MBUF0,R5       ;DID SRC REG GET AUTO-DECREMENTED?
14470  054136  001401                        BEQ     TST735          ;;BR IF YES
14471
14472  054140  104046                        ERROR   46              ;AUTO DECREMENT DID NOT OCCUR
14473
14474                          ;;**********************************************************************
14475                          ;*TEST 735      MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
14476                          ;;**********************************************************************
```

```
14477  054142                          TST735:
14478  054142  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14479  054144  012700  000735          MOV     #735,R0                 ;;LOAD R0 WITH TEST NUMBER
14480  054150  013737  054210  001076  MOV     @#2$,@#STMP0            ;GET TEST INSTRUCTION WORD
14481  054156  005001                  CLR     R1                      ;S/B RESULT IN R2
14482  054160  012704  106420          MOV     #106420,R4              ;/S/B RESULT IN R3
14483  054164  012702  177776          MOV     #-2,R2                  ;INITIALIZE REG
14484  054170  012703  177777          MOV     #-1,R3                  ;INITIALIZE REG + 1
14485  054174  012705  063310          MOV     #ATA+12,R5              ;SET UP POINTER TO POINTER TO MBUF0
14486  054200  012737  134570  063312  MOV     #-43210,@#MBUF0         ;INITIALIZE SRC
14487  054206  000277                  SCC                             ;SCOPE SYNC
14488
14489  054210  070255          2$:     MUL     @-(R5),R2               ;TEST THE MUL
14490
14491  054212  100403                  BMI     3$                      ;N:C=0001?
14492  054214  001402                  BEQ     3$
14493  054216  102401                  BVS     3$
14494  054220  103401                  BCS     4$
14495
14496  054222  104044          3$:     ERROR   44                      ;COND CODES SET IMPROPERLY
14497
14498  054224  020304          4$:     CMP     R3,R4                   ;REG+1 CORRECT?
14499  054226  001002                  BNE     5$                      ;BR IF NOT
14500  054230  020102                  CMP     R1,R2                   ;REG CORRECT?
14501  054232  001401                  BEQ     6$                      ;BR IF YES
14502
14503  054234  104045          5$:     ERROR   45                      ;MUL DELIVERED WRONG RESULT
14504
14505  054236  022705  063306  6$:     CMP     #ATA+10,R5              ;DID R5 GET AUTO-DECREMENTED?
14506  054242  001401                  BEQ     TST736                  ;;BR IF YES
14507
14508  054244  104046                  ERROR   46                      ;AUTO INCREMENT DID NOT OCCUR
14509
14510                                  ;;*****************************************************************
14511                                  ;*TEST 736       MUL X(RA),RB TEST ; N:C = 1111 TO 0100
14512                                  ;;*****************************************************************
14513  054246                          TST736:
14514  054246  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14515  054250  012700  000736          MOV     #736,R0                 ;;LOAD R0 WITH TEST NUMBER
14516  054254  013737  054310  001076  MOV     @#2$,@#STMP0            ;GET TEST INSTRUCTION WORD
14517  054262  005001                  CLR     R1                      ;S/B RESULT IN R2
14518  054264  005004                  CLR     R4                      ;S/B RESULT IN R3
14519  054266  012702  012345          MOV     #012345,R2              ;INITIALIZE REG
14520  054272  012703  177777          MOV     #-1,R3                  ;INITIALIZE REG + 1
14521  054276  012705  063312          MOV     #MBUF0,R5               ;SET UP POINTER TO SRC
14522  054302  005065  000002          CLR     2(R5)                   ;INITIALIZE SRC
14523  054306  000277                  SCC                             ;SCOPE SYNC
14524
14525  054310  070265  000002  2$:     MUL     2(R5),R2                ;TEST THE MUL
14526
14527  054314  100403                  BMI     3$                      ;N:C=0100?
14528  054316  001002                  BNE     3$
14529  054320  102401                  BVS     3$
14530  054322  103001                  BCC     4$
14531
14532  054324  104044          3$:     ERROR   44                      ;COND CODES SET IMPROPERLY
```

```
14533
14534  054326  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14535  054330  001002                          BNE     5$              ;BR IF NOT
14536  054332  020102                          CMP     R1,R2           ;REG CORRECT?
14537  054334  001401                          BEQ     TST737          ;;BR IF YES
14538
14539  054336  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14540
14541                                  ;;**************************************************************
14542                                  ;*TEST 737       MUL @X(RA),R0 TEST
14543                                  ;;**************************************************************
14544  054340                          TST737:
14545  054340  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14546  054342  012700  000737                  MOV     #737,R0         ;;LOAD R0 WITH TEST NUMBER
14547  054346  013737  054404  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14548  054354  005001                          CLR     R1              ;S/B RESULT IN R2
14549  054356  012704  000100                  MOV     #100,R4         ;S/B RESULT IN R3
14550  054362  012702  000010                  MOV     #10,R2          ;INITIALIZE REG
14551  054366  005003                          CLR     R3              ;INITIALIZE REG + 1
14552  054370  012705  063276                  MOV     #ATA,R5         ;GET POINTER TO TABLE OF POINTERS
14553  054374  012737  000010  063312          MOV     #10,@#MBUF0     ;INITIALIZE SRC
14554  054402  000257                          CCC                     ;SCOPE SYNC
14555
14556  054404  070275  000010          2$:     MUL     @10(R5),R2      ;TEST THE MUL
14557
14558  054410  020304                          CMP     R3,R4           ;REG+1 CORRECT?
14559  054412  001002                          BNE     3$              ;BR IF NOT
14560  054414  020102                          CMP     R1,R2           ;REG CORRECT?
14561  054416  001401                          BEQ     TST740          ;;BR IF YES
14562
14563  054420  104045                  3$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14564
14565                                  ;;**************************************************************
14566                                  ;*TEST 740       DIV #N,RA TEST ; N:C = 1111
14567                                  ;;**************************************************************
14568  054422                          TST740:
14569  054422  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14570  054424  012700  000740                  MOV     #740,R0         ;;LOAD R0 WITH TEST NUMBER
14571  054430  013737  054456  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
14572  054436  012701  010000                  MOV     #010000,R1      ;S/B RES IN R2
14573  054442  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14574  054446  005002                          CLR     R2              ;SET UP REG OPERAND
14575  054450  012703  020001                  MOV     #020001,R3      ;SET UP REG+1 OP
14576  054454  000277                          SCC                     ;SCOPE SYNC
14577
14578  054456  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14579
14580  054462  100403                          BMI     3$              ;N:C=0000?
14581  054464  001402                          BEQ     3$
14582  054466  102401                          BVS     3$
14583  054470  103001                          BCC     4$
14584
14585  054472  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14586
14587  054474  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14588  054476  001002                          BNE     5$              ;BR IF NOT
```

```
14589  054500  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14590  054502  001401                          BEQ     TST741          ;;BR IF YES
14591
14592  054504  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14593
14594                                  ;;**********************************************************************
14595                                  ;*TEST 741      DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
14596                                  ;;**********************************************************************
14597  054506                          TST741:
14598  054506  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14599  054510  012700  000741                  MOV     #741,R0         ;;LOAD R0 WITH TEST NUMBER
14600  054514  013737  054544  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
14601  054522  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14602  054526  012704  177776                  MOV     #-2,R4          ;S/B RES IN R3
14603  054532  012702  177777                  MOV     #-1,R2          ;SET UP REG OPERAND
14604  054536  012703  177762                  MOV     #-14.,R3        ;SET UP REG+1 OP
14605  054542  000257                          CCC                     ;SCOPE SYNC
14606
14607  054544  071227  000004          2$:     DIV     #4,R2           ;TEST DIV
14608
14609  054550  100003                          BPL     3$              ;N:C=1000?
14610  054552  001402                          BEQ     3$
14611  054554  102401                          BVS     3$
14612  054556  103001                          BCC     4$
14613
14614  054560  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14615
14616  054562  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14617  054564  001002                          BNE     5$              ;BR IF NOT
14618  054566  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14619  054570  001401                          BEQ     TST742          ;;BR IF YES
14620
14621  054572  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14622
14623                                  ;;**********************************************************************
14624                                  ;*TEST 742      DIV #N,RA TEST ; N:C = 0000 TO 0100
14625                                  ;;**********************************************************************
14626  054574                          TST742:
14627  054574  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14628  054576  012700  000742                  MOV     #742,R0         ;;LOAD R0 WITH TEST NUMBER
14629  054602  013737  054626  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
14630  054610  005001                          CLR     R1              ;S/B RES IN R2
14631  054612  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14632  054616  005002                          CLR     R2              ;SET UP REG OPERAND
14633  054620  012703  000001                  MOV     #1,R3           ;SET UP REG+1 OP
14634  054624  000257                          CCC                     ;SCOPE SYNC
14635
14636  054626  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14637
14638  054632  100403                          BMI     3$              ;N:C=0100?
14639  054634  001002                          BNE     3$
14640  054636  102401                          BVS     3$
14641  054640  103001                          BCC     4$
14642
14643  054642  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14644
```

```
14645  054644  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14646  054646  001002                          BNE     5$              ;BR IF NOT
14647  054650  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14648  054652  001401                          BEQ     TST743          ;;BR IF YES
14649
14650  054654  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14651
14652                                  ;;**********************************************************************
14653                                  ;*TEST 743      DIV #-N,RA TEST ; RA POS
14654                                  ;;**********************************************************************
14655  054656                          TST743:
14656  054656  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14657  054660  012700  000743                  MOV     #743,R0         ;;LOAD R0 WITH TEST NUMBER
14658  054664  013737  054712  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
14659  054672  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14660  054676  012704  000002                  MOV     #2,R4           ;S/B RES IN R3
14661  054702  005002                          CLR     R2              ;SET UP REG OPERAND
14662  054704  012703  000016                  MOV     #14.,R3         ;SET UP REG+1 OP
14663  054710  000257                          CCC                     ;SCOPE SYNC
14664
14665  054712  071227  177774          2$:     DIV     #-4,R2          ;TEST DIV
14666
14667  054716  020304                          CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14668  054720  001002                          BNE     3$              ;BR IF NOT
14669  054722  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14670  054724  001401                          BEQ     TST744          ;;BR IF YES
14671
14672  054726  104045                  3$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14673
14674                                  ;;**********************************************************************
14675                                  ;*TEST 744      DIV  TEST - V BIT GETS SET
14676                                  ;*      THIS TEST TESTS THAT THE V BIT CAN BE SET IN ALL THE
14677                                  ;*      POSSIBLE WAYS.  SINCE THE INSTRUCTION SHOULD BE ABORTED, THE
14678                                  ;*      RESULTS CANNOT BE GUARANTEED.  FOR THIS REASON, ONLY
14679                                  ;*      THE CONDITION CODES ARE CHECKED.
14680                                  ;;**********************************************************************
14681  054730                          TST744:
14682  054730  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14683  054732  012700  000744                  MOV     #744,R0         ;;LOAD R0 WITH TEST NUMBER
14684  054736  013701  054756                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14685  054742  012704  000002                  MOV     #2,R4           ;S/B PSW
14686  054746  005037  177776                  CLR     @#PSW           ;CLEAR OUT OTHER PSW BITS
14687  054752  012702  000050                  MOV     #50,R2          ;SET UP REG OP
14688
14689  054756  071227  000005          2$:     DIV     #5,R2           ;TEST DIV -- SHOULD ABORT
14690
14691  054762  100424                          BMI     3$              ;N:C=0010?
14692  054764  001423                          BEQ     3$
14693  054766  102022                          BVC     3$
14694  054770  103421                          BCS     3$
14695
14696  054772  012702  177777                  MOV     #-1,R2          ;INITIALIZE REG OP
14697  054776  005003                          CLR     R3              ;INITILAIZE REG+1 OP
14698
14699  055000  071227  177776                  DIV     #-2,R2          ;TEST DIV -- SHOULD ABORT
14700
```

```
14701  055004  100413                        BMI     3$              ;N:C=0010?
14702  055006  001412                        BEQ     3$
14703  055010  102011                        BVC     3$
14704  055012  103410                        BCS     3$
14705
14706  055014  012704  000003                MOV     #3,R4           ;S/B PSW
14707
14708  055020  071227  000000                DIV     #0,R2           ;TEST DIV BY 0 -- SHOULD ABORT
14709
14710  055024  100403                        BMI     3$              ;N:C=0010?
14711  055026  001402                        BEQ     3$
14712  055030  102001                        BVC     3$
14713  055032  103405                        BCS     TST745          ;;IF ALL OK, THEN EXIT TEST
14714
14715  055034  013703  177776        3$:     MOV     @#PSW,R3        ;GET WAS PSW
14716  055040  012702  177776                MOV     #PSW,R2         ;DESTINATION IS PSW
14717
14718  055044  104001                        ERROR   1               ;CONDITION CODES SET WRONG
14719
14720                                ;;**********************************************************************
14721                                ;*TEST 745      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14722                                ;;**********************************************************************
14723  055046                        TST745:
14724  055046  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14725  055050  012700  000745                MOV     #745,R0         ;;LOAD R0 WITH TEST NUMBER
14726  055054  013701  055072                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14727  055060  012704  123450                MOV     #123450,R4      ;S/B RESULT
14728  055064  012703  112345                MOV     #112345,R3      ;INITIAL REG
14729  055070  000257                        CCC                     ;SCOPE SYNC
14730
14731  055072  072327  000003        2$:     ASH     #3,R3           ;TEST THE ASH
14732
14733  055076  100003                        BPL     3$              ;N:C=1010?
14734  055100  001402                        BEQ     3$
14735  055102  102001                        BVC     3$
14736  055104  103001                        BCC     4$
14737
14738  055106  104002                3$:     ERROR   2               ;INCORRECT CONDITION CODES
14739
14740  055110  020304                4$:     CMP     R3,R4           ;CORRECT RESULT?
14741  055112  001401                        BEQ     TST746          ;;BR IF YES
14742  055114  104002                        ERROR   2               ;ASH DELIVERED WRONG RESULT
14743
14744                                ;;**********************************************************************
14745                                ;*TEST 746      ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14746                                ;;**********************************************************************
14747  055116                        TST746:
14748  055116  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14749  055120  012700  000746                MOV     #746,R0         ;;LOAD R0 WITH TEST NUMBER
14750  055124  013701  055142                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14751  055130  005004                        CLR     R4              ;S/B RESULT
14752  055132  012703  000004                MOV     #4,R3           ;INITIAL REG
14753  055136  000257                        CCC                     ;SCOPE SYNC
14754  055140  000270                        SEN                     ;CODES = 1000
14755
14756  055142  072327  177775        2$:     ASH     #-3,R3          ;TEST THE ASH
```

```
14757
14758  055146  100403                          BMI     3$              ;N:C=0101?
14759  055150  001002                          BNE     3$
14760  055152  102401                          BVS     3$
14761  055154  103401                          BCS     4$
14762
14763  055156  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14764
14765  055160  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14766  055162  001401                          BEQ     TST747          ;;BR IF YES
14767  055164  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14768
14769                                  ;;*****************************************************************
14770                                  ;*TEST 747      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14771                                  ;;*****************************************************************
14772  055166                          TST747:
14773  055166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14774  055170  012700  000747                  MOV     #747,R0         ;;LOAD R0 WITH TEST NUMBER
14775  055174  013701  055212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14776  055200  012704  177234                  MOV     #177234,R4      ;S/B RESULT
14777  055204  012703  123432                  MOV     #123432,R3      ;INITIAL REG
14778  055210  000277                          SCC                     ;SCOPE SYNC
14779
14780  055212  072327  177772          2$:     ASH     #-6,R3          ;TEST THE ASH
14781
14782  055216  100003                          BPL     3$              ;N:C=1000?
14783  055220  001402                          BEQ     3$
14784  055222  102401                          BVS     3$
14785  055224  103001                          BCC     4$
14786
14787  055226  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14788
14789  055230  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14790  055232  001401                          BEQ     TST750          ;;BR IF YES
14791  055234  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14792
14793                                  ;;*****************************************************************
14794                                  ;*TEST 750      ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14795                                  ;;*****************************************************************
14796  055236                          TST750:
14797  055236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14798  055240  012700  000750                  MOV     #750,R0         ;;LOAD R0 WITH TEST NUMBER
14799  055244  013737  055274  001076          MOV     @#2$,@#TMP0     ;GET TEST INSTRUCTION WORD
14800  055252  012701  123456                  MOV     #123456,R1      ;S/B RES IN R2
14801  055256  012704  076530                  MOV     #076530,R4      ;S/B RES IN R3
14802  055262  012702  112345                  MOV     #112345,R2      ;INITIALIZE COMBINED
14803  055266  012703  147653                  MOV     #147653,R3      ;  REGISTERS
14804  055272  000257                          CCC                     ;SCOPE SYNC
14805
14806  055274  073227  000003          2$:     ASHC    #3,R2           ;TEST ASHC
14807
14808  055300  100003                          BPL     3$              ;N:C=1010?
14809  055302  001402                          BEQ     3$
14810  055304  102001                          BVC     3$
14811  055306  103001                          BCC     4$
14812
```

```
14813  055310  104044                    3$:     ERROR   44              ;COND CODES WRONG
14814
14815  055312  020102                    4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14816  055314  001002                            BNE     5$              ;BR IF NOT
14817  055316  020403                            CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14818  055320  001401                            BEQ     TST751          ;BR IF YES
14819  055322  104045                    5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14820
14821                                    ;;**********************************************************************
14822                                    ;*TEST 751      ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14823                                    ;;**********************************************************************
14824  055324                            TST751:
14825  055324  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14826  055326  012700  000751                    MOV     #751,R0         ;;LOAD R0 WITH TEST NUMBER
14827  055332  013737  055356  001076            MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14828  055340  005001                            CLR     R1              ;S/B RES IN R2
14829  055342  005004                            CLR     R4              ;S/B RES IN R3
14830  055344  005002                            CLR     R2              ;INITIALIZE COMBINED
14831  055346  012703  000005                    MOV     #5,R3           ;  REGISTERS
14832  055352  000257                            CCC                     ;SCOPE SYNC
14833  055354  000270                            SEN                     ;CODES = 1000
14834
14835  055356  073227  177775            2$:     ASHC    #-3,R2          ;TEST ASHC
14836
14837  055362  100403                            BMI     3$              ;N:C=0101?
14838  055364  001002                            BNE     3$
14839  055366  102401                            BVS     3$
14840  055370  103401                            BCS     4$
14841
14842  055372  104044                    3$:     ERROR   44              ;COND CODES WRONG
14843
14844  055374  020102                    4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14845  055376  001002                            BNE     5$              ;BR IF NOT
14846  055400  020403                            CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14847  055402  001401                            BEQ     TST752          ;;BR IF YES
14848  055404  104045                    5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14849
14850                                    ;;**********************************************************************
14851                                    ;*TEST 752      ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1111 TO 1000
14852                                    ;;**********************************************************************
14853  055406                            TST752:
14854  055406  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14855  055410  012700  000752                    MOV     #752,R0         ;;LOAD R0 WITH TEST NUMBER
14856                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT15
14857  055414  032737  100000  063234            BIT     #BIT15,@#BPTLOC ;BREAKPOINT HALT SET ??
14858  055422  001401                            BEQ     .+4             ;BR IF NOT
14859  055424  000000                            HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
14860  055426  013737  055456  001076            MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
14861  055434  012701  177234                    MOV     #177234,R1      ;S/B RES IN R2
14862  055440  012704  135275                    MOV     #135275,R4      ;S/B RES IN R3
14863  055444  012702  123456                    MOV     #123456,R2      ;INITIALIZE COMBINED
14864  055450  012703  127542                    MOV     #127542,R3      ;  REGISTERS
14865  055454  000257                            CCC                     ;SCOPE SYNC
14866
14867  055456  073227  177772            2$:     ASHC    #-6,R2          ;TEST ASHC
14868
```

```
14869  055462  100003                      BPL     3$              ;N:C=1000?
14870  055464  001402                      BEQ     3$
14871  055466  102401                      BVS     3$
14872  055470  103401                      BCS     4$
14873
14874  055472  104044              3$:     ERROR   44              ;COND CODES WRONG
14875
14876  055474  020102              4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14877  055476  001002                      BNE     5$              ;BR IF NOT
14878  055500  020403                      CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14879  055502  001401                      BEQ     TST753          ;;BR IF YES
14880  055504  104045              5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14881
14882                                       ;*
14883                                       ;*      THIS SECTION OF THE MED TESTS EXERCISES CERTAIN SCRATCH
14884                                       ;*      PAD REGISTERS USING MED READS AND WRITES.  THEIR ORIGINAL
14885                                       ;*      CONTENTS ARE RESTORED BUT:
14886                                       ;*
14887                                       ;*      ******************** IMPORTANT NOTE ********************
14888                                       ;*      *                                                      *
14889                                       ;*      *  THE CONSOLE MUST NOT !!! BE USED DURING THESE MED   *
14890                                       ;*      *  TESTS. NO INTERRUPTS OR TRAPS CAN BE ALLOWED EITHER *
14891                                       ;*      *                                                      *
14892                                       ;*      ********************************************************
14893                                       ;*
14894
14895                                       ;;*********************************************************************
14896                                       ;*TEST 753      CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
14897                                       ;*      THE NEXT TEST BELOW CHECKS TO SEE THAT THE "MED"
14898                                       ;*      (MAINTENANCE, EXAM, AND DEPOSIT) INSTRUCTION WILL EXECUTE
14899                                       ;*      WHEN IN KERNEL MODE WITHOUT AFFECTING THE PSW AND
14900                                       ;*      THAT IT IS ILLEGAL IN USER MODE
14901                                       ;;*********************************************************************
14902  055506                               TST753:
14903  055506  012700  000752                      MOV     #752,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14904  055512  000004                      SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14905  055514  012737  000304  177770      MED1:   MOV     #304,@#UBREAK   ;SET SCOPE SYNC FOR MED INSTR
14906  055522  012737  140000  177776              MOV     #140000,@#PSW   ;GO TO USER MODE
14907  055530  012706  001000                      MOV     #STACK,SP       ;SETUP U  R STACK PTR.
14908  055534  012737  055566  000004              MOV     #2$,@#ERRVEC    ;SET EP  S TRAP VECTOR TO 2$ BELOW
14909  055542  012737  055566  000010              MOV     #2$,@#RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
14910  055550  012701  177777                      MOV     #-1,R1          ;LOAD R1 WITH A -1
14911  055554  005000                      CLR     R0                      ;CLEAR R0
14912  055556  076600                      MED                             ;TRY TO DO MAINT. EXAMINE
14913  055560  000041                      .WORD   041                     ;MED READ CODE FOR R1
14914  055562  104012                      ERROR   12                      ;ERROR - MED INST. NOT ILLEGAL IN USER
14915  055564  000404                      BR      4$
14916  055566  005700              2$:     TST     R0                      ;IS R0 UNCHANGED?
14917  055570  001401                      BEQ     3$                      ;BRANCH IF YES
14918  055572  104013                      ERROR   13                      ;ERROR - MED INSTRUCTION WAS EXECUTED
14919                                                                       ;BEFORE TRAPPING
14920  055574  022626              3$:     CMP     (SP)+,(SP)+             ;CLEAN UP STACK
14921  055576  012737  061220  000004  4$:  MOV     #BERR,@#ERRVEC  ;RESTORE ERROR TRAP VECTOR
14922  055604  012737  061122  000010              MOV     #RSERR,@#RESVEC ;RESTORE RESERVED INST. TRAP VECTOR
14923
14924  055612  005037  177776      MED0:   CLR     @#PSW                   ;GO TO KERNEL MODE,CLEAR COND. CODES
```

```
14925  0[illegible]  076600                    MED                     ;DO MAINT. EXAMINE OF R1
14926  0[illegible]  000041                    .WORD   041             ;MED READ CODE FOR R1
14927  0[illegible]  103403                    BCS     MEDHLT
14928  0[illegible]  103402                    BVS     MEDHLT
14929  0[illegible]  100401                    BMI     MEDHLT
14930  055630  001001                          BNE     .+4
14931  055632  104014            MEDHLT:       ERROR   14              ;ERROR CC-BITS IN PSW AFFECTED BY MED
14932
14933                                          ;;*****************************************************************
14934                                          ;*TEST 754      MED TEST - R/W DATA PATTERNS TO REGS
14935                                          ;*      THIS PARTICULAR MED TEST WRITES DATA PATTERNS
14936                                          ;*      TO THOSE INTERNAL REGS. WHICH CAN BE WRITTEN
14937                                          ;*      AND READ WITHOUT SPECIAL CONSIDERAIONS.  REGISTERS
14938                                          ;*      REQUIRING SPECIAL TESTS ARE TESTED IN LATER
14939                                          ;*      MED TESTS.
14940                                          ;*      TABLE II CONTAINS THE REGISTER ADDRESSES.
14941                                          ;*
14942                                          ;*      A MAX. OF 3 ERRORS ARE REPORTED FOR EACH LOC.
14943                                          ;*
14944                                          ;;*****************************************************************
14945  055634                                  TST754:
14946  055634  012700  000753                 MOV     #753,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14947  055640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14948  055642  012737  000340  177776  MEDT1:  MOV     #340,@#PSW      ;KERNEL MODE-PRIORITY 7
14949  055650  012701  064166                 MOV     #TBL2,R1        ;INITIALIZE ADDRESS POINTER
14950  055654  012737  125252  001102  1$:     MOV     #125252,@#TMP2
14951  055662  111137  055730                 MOVB    (R1),@#11$      ;PUT WRITE CODE BY "WRITE-MED'S"
14952  055666  112137  055752                 MOVB    (R1)+,@#13$     ;AND POINT R1 TO READ CODE
14953  055672  111137  055710                 MOVB    (R1),@#10$      ;PUT READ CODE BY "READ-MED'S"
14954  055676  112137  055736                 MOVB    (R1)+,@#12$     ;R1 NOW POINTS TO NEXT REG.
14955  055702  005037  001106                 CLR     @#TMP4          ;CLEAR ERROR COUNTER
14956  055706  076600            2$:          MED                     ;MED-READ THE INTERNAL REG.
14957  055710  000000            10$:         .WORD   0               ;MED-READ CODE
14958  055712  010037  001076                 MOV     R0,@#TMP0       ;SAVE ITS ORIGINAL CONTENTS
14959  055716  010137  001100                 MOV     R1,@#TMP1       ;SAVE ADDR. PTR. VALUE
14960  055722  013700  001102                 MOV     @#TMP2,R0       ;LOAD R0 WITH DATA TO BE WRITTEN
14961  055726  076600                          MED                     ;MED-WRITE THE TEST DATA
14962  055730  000000            11$:         .WORD   0               ;MED-WRITE CODE
14963  055732  005000                          CLR     R0              ;CLEAR R0
14964  055734  076600                          MED                     ;MED-READ THE DATA BACK
14965  055736  000000            12$:         .WORD   0               ;MED-READ CODE
14966  055740  010037  001104                 MOV     R0,@#TMP3       ;SAVE DATA READ FOR COMPARISON
14967  055744  013700  001076                 MOV     @#TMP0,R0       ;LOAD ORIGINAL DATA IN R0
14968  055750  076600                          MED                     ;MED-WRITE ORG. DATA TO REG.
14969  055752  000000            13$:         .WORD   0               ;MED-WRITE CODE
14970  055754  023737  001102  001104         CMP     @#TMP2,@#TMP3   ;DID DATA READ=DATA WRITTEN?
14971  055762  001412                          BEQ     3$              ;BRANCH IF YES
14972  055764  013737  055736  001100         MOV     @#12$,@#TMP1    ;SAVE MED-CODE FOR ERROR
14973  055772  022737  000003  001106         CMP     #3,@#TMP4       ;MAX. ERROR REPORTS YET?
14974  056000  002401                          BLT     14$             ;BRANCH IF YES
14975  056002  104022                          ERROR   22              ;INT. REG. READ BACK WRONG DATA
14976  056004  005237  001106            14$:  INC     @#TMP4          ;INCREMENT ERROR COUNTER
14977  056010  005137  001102            3$:   COM     @#TMP2          ;CHANGE DATA PATTERN
14978  056014  013701  001100                  MOV     @#TMP1,R1       ;RESTORE ADDR. POINTER
14979  056020  022737  125252  001102         CMP     #125252,@#TMP2  ;BOTH DATA PATTERNS BEEN USED?
14980  056026  001327                          BNE     2$              ;BRANCH IF NO
```

```
14981  056030  005711                          TST     (R1)            ;END OF ADDR. TABLE?
14982  056032  001310                          BNE     1$              ;BRANCH IF NO
14983
14984                                  ;;*****************************************************************
14985                                  ;*TEST 755      MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14986                                  ;*
14987                                  ;*      THIS TEST CHECKS ALL OF THE "NOP" OPERATION CODES
14988                                  ;*      TO ENSURE THEY WILL EXECUTE AS NOP'S AND
14989                                  ;*      NOT RESULT IN A PROCESSOR HANG.  THE "NOPS"
14990                                  ;*      TABLE (TABLE III) HOLDS THESE CODES.
14991                                  ;*      THIS TEST ALSO READS THE PROGRAM COUNTER (R7) VALUES
14992                                  ;*      STORED IN A & B SCRATCH PADS TO SEE THAT THEY
14993                                  ;*      READ PROPERLY.  THE R7 ADDRESSES ARE IN TABLE IV.
14994                                  ;*
14995                                  ;;*****************************************************************
14996  056034                          TST755:
14997  056034  012700  000754                 MOV     #754,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14998  056040  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14999  056042  012701  064404         MEDT3:  MOV     #TBL3,R1        ;INITIALIZE NOP TABLE PTR. (R1)
15000  056046  112137  056054         1$:     MOVB    (R1)+,@#10$     ;PLACE FIRST "NOP-CODE" AFTER MED
15001                                                                   ;AND POINT R1 TO LAST CODE IN GROUP
15002  056052  076600                 5$:     MED                     ;EXECUTE MED WITH NOP OP-CODE
15003  056054  000000                 10$:    .WORD   0
15004  056056  123711  056054                 CMPB    @#10$,(R1)      ;HAVE ALL NOPS IN THAT GROUP
15005                                                                   ;BEEN TESTED?
15006  056062  103003                          BHIS    6$              ;BRANCH IF YES
15007  056064  005237  056054                 INC     @#10$           ;NEXT NOP IN GROUP
15008  056070  000770                          BR      5$
15009  056072  105721                 6$:     TSTB    (R1)+           ;POINT R1 TO NEXT NOP GROUP
15010  056074  005711                          TST     (R1)            ;HAVE ALL GROUPS BEEN TESTED
15011  056076  001363                          BNE     1$              ;BRANCH IF NO
15012
15013  056100  113737  064425  056112 MEDT4:  MOVB    @#R7A+1,@#5$    ;LOAD R7A READ CODE AFTER MED
15014  056106  005000                 4$:     CLR     R0              ;CLEAR R0
15015  056110  076600                          MED                     ;MED READ R7 IN THE ASP
15016  056112  000000                 5$:     .WORD   0               ;READ CODE FOR R7A
15017  056114  027027  056114                 CMP     R0,#5$+2        ;DID R7A READ CORRECTLY?
15018  056120  001411                          BEQ     6$              ;BRANCH IF YES
15019  056122  013737  056112  001100        MOV     @#5$,@#TMP1     ;SAVE MED-CODE FOR ERROR
15020  056130  012737  056114  001102        MOV     #5$+2,@#TMP2    ;SAVE DATA EXPECTED
15021  056136  010037  001104                MOV     R0,@#TMP3       ;SAVE DATA RECEIVED
15022  056142  104022                          ERROR   22              ;R7A DID NOT READ THE RIGHT VALUE
15023  056144  023727  056112  000047 6$:     CMP     @#5$,#47        ;HAS R7B BEEN CHECKED?
15024  056152  001404                          BEQ     8$              ;BRANCH IF YES
15025  056154  113737  064431  056112        MOVB    @#R7B+1,@#5$    ;LOAD R7B READ CODE AFTER MED
15026  056162  000751                          BR      4$              ;TEST R7 BSP
15027  056164                          8$:
15028
15029
15030                                  ;;*****************************************************************
15031                                  ;*TEST 756      MED TEST - CSP CONSTANTS CHECK
15032                                  ;*
15033                                  ;*      THIS TEST CHECKS THE CONSTANT VALUES LOCATED
15034                                  ;*      IN THE C SCRATCH PAD.  THE CONSTANTS ARE READ
15035                                  ;*      WITH A MED INSTRUCTION AND COMPARED TO THEIR
15036                                  ;*      EXPECTED VALUE.  THE ADDRESSES OF THESE CONSTANTS
```

```
15037                                   ;*      AND THE VALUES EXPECTED ARE IN TABLE VII.
15038                                   ;*
15039                                   ;;*****************************************************************
15040  056164                           TST756:
15041  056164  012700  000755                   MOV     #755,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15042  056170  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15043
15044  056172  076600                           MED
15045  056174  000144                           RDFLAG
15046  056176  052700  004000                   BIS     #BIT11,R0       ;SET THE "CSP INVALID BIT" IN FLAG REG.
15047  056202  076600                           MED
15048  056204  000154                           WRFLAG
15049  056206  170000                  MED10:   CFCC                    ;EXECUTE FLT. PT INST. SO FLT. PT.
15050                                           ;CONSTANTS ARE LOADED INTO CSP
15051  056210  012701  064536                   MOV     #TBL7,R1        ;SETUP TABLE POINTER
15052  056214  012167  000006          10$:     MOV     (R1)+,1$        ;LOAD MED READ CODE AT 1$
15053  056220  001414                           BEQ     11$             ;BR IF END OF TABLE
15054  056222  005000                           CLR     R0
15055  056224  076600                           MED                     ;READ INTERNAL CONTENTS INTO R0
15056  056230  000000                  1$:      .WORD   0
15057  056232  020021                           CMP     R0,(R1)+        ;WAS THE CONSTANT READ THE ONE EXPECTED
15058  056234  001770                           BEQ     10$             ;BRANCH IF YES
15059  056236  013737  056226  001100           MOV     @#1$,@#$TMP1    ;SAVE MEDCODE FOR ERROR
15060  056242  016137  177776  001102           MOV     -2(R1),@#$TMP2  ;SAVE CONSTANT VALUE EXPECTED
15061  056250  104021                           ERROR   21              ;CSP LOCATION HELD WRONG VALUE
15062  056252                          11$:
15063
15064
15065                                   ;;*****************************************************************
15066                                   ;*TEST 757      MED TEST - MICROBK CHECK OF MICRO-POINTS
15067                                   ;*
15068                                   ;*      THIS TEST USES THE MICROBREAK REGISTER AND THE
15069                                   ;*      INFORMATION IN TABLE V TO CHECK THAT THE
15070                                   ;*      CORRECT MED-FLOW IS ENTERED WHEN EACH
15071                                   ;*      REGISTER IS ACCESSED BY A MED INSTRUCTION.
15072                                   ;*      THE MICROBREAK REG. IS SETUP TO CAUSE A TRAP TO
15073                                   ;*      LOC. 4 WHEN ITS CONTENTS EQUAL THE ADDRESS
15074                                   ;*      OF THE MIRCOWORD BEING EXECUTED.
15075                                   ;*
15076                                   ;*          NOTE:  THE MICRO BREAK - TRAP-TO-4 CAPABILITY
15077                                   ;*                 IS TRIED AT THE BEGINNING OF THE TEST.
15078                                   ;*                 IF IT DOESN'T WORK, AN ERROR IS PRINTED
15079                                   ;*                 AND THE TEST IS SKIPPED
15080                                   ;*
15081                                   ;;*****************************************************************
15082  056252                           TST757:
15083  056252  012700  000756                   MOV     #756,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15084  056256  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15085  056260  012737  000071  177770  MED11:   MOV     #SWB01,@#UBREAK ;LOAD MICROBK. REG. WITH AN MICRO ADDR.
15086  056266  012737  061104  000004           MOV     #BKROUT,@#4     ;LOAD ADDR. OF MICROBK. ROUTINE IN 4
15087  056274  012737  000340  000006           MOV     #340,@#6        ;LOAD KERNEL PSW - PRIORITY 7 IN 6
15088  056302  005037  061112                   CLR     @#BKFLAG        ;CLEAR MICROBK. TRAP FLAG
15089  056306  076600                           MED                     ;GET WHAMI INTO R0
15090  056310  000022                           RDWHAMI
15091  056312  052700  001000                   BIS     #BIT9,R0        ;SET BIT 9
15092  056316  076600                           MED                     ;MED-WRITE THE WHAMI REG TO
```

```
15093  056320  000222                  10$:    WRWHAMI                 ;ENABLE MICROBK-TRAP-TO-4
15094  056322  076600                          MED                     ;GET FLAG REGISTER
15095  056324  000144                          RDFLAG
15096  056326  052700  100000                  BIS     #BIT15,R0       ;SET BIT 15 IN R0
15097  056332  076600                          MED                     ;MED-WRITE THE FLAG REG TO
15098  056334  000344                  11$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15099  056336  010000                          SWAB    R0              ;MICROBK TRAP SHOULD OCCUR ON SWAB
15100  056340  005737  061112                  TST     @#BKFLAG        ;DID TRAP TO 4 OCCUR?
15101  056344  001007                          BNE     1$              ;BRANCH IF YES
15102  056346  005037  001076                  CLR     @#STMP0
15103  056352  016737  121513  001100          MOV     SWB01,@#STMP1   ;SAVE EXPECTED UBREAK ADDR
15104  056360  104015                          ERROR   15              ;MICROBK TRAP DIDN'T WORK
15105  056362  000453                          BR      50$             ;SKIP TO END OF TEST
15106
15107  056364  012701  000710          1$:     MOV     #SWB01+10,R1    ;GET CORRECT U-ADDR
15108  056370  076600                          MED                     ;GET LOG CUA REG
15109  056372  000103                          RDLCUA
15110  056374  042700  100007                  BIC     #100007,R0      ;GET RID OF IRRELEVANT BITS
15111  056400  020001                          CMP     R0,R1           ;WAS CORRECT UADDR LOGGED?
15112  056402  001401                          BEQ     3$              ;BR IF YES
15113  056404  104025                          ERROR   25              ;CUA CONTAINS INCORRECT U-ADDR
15114  056406  012701  064436          3$:     MOV     #TBL5,R1                ;INTIALIZE TABLE PTR. (R1)
15115  056412  012702  064464                  MOV     #TBL6,R2
15116  056416  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP RETURN TO 2$
15117  056422  111137  056460          2$:     MOVB    (R1),@#12$      ;LOAD WRITE CODE AFTER MED
15118  056426  001431                          BEQ     50$             ;BR IF END OF TABLE
15119  056430  011237  177770          4$:     MOV     (R2),@#UBREAK   ;LOAD MICROBK REG. WITH MICROADDR.
15120  056434  005037  061112                  CLR     @#BKFLAG        ;CLEAR MICROBK TRAP-TO-4 FLAG
15121  056440  076600                          MED                     ;GET FLAG REGISTER
15122  056442  000144                          RDFLAG
15123  056444  052700  100000                  BIS     #BIT15,R0       ;SET BIT 15 IN R0
15124  056450  076600                          MED                     ;MED WRITE TO FLAG REG TO
15125  056452  000344                  15$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15126  056454  005000                          CLR     R0              ;IN CASE U-BREAK TRAP DOESN'T OCCOR
15127                                                                  ;USUALLY BETTER TO WRITE 0'S
15128  056456  076600                          MED
15129  056460  000000                  12$:    .WORD   0
15130  056462  005737  061112                  TST     @#BKFLAG        ;DID WE TRAP-TO-4? (FLAG NOT = 0)
15131  056466  001006                          BNE     20$             ;BRANCH IF YES TO NEXT ENTRY
15132  056470  013737  056460  001076          MOV     @#12$,@#STMP0   ;SAVE MED-CODE FOR ERROR
15133  056476  011237  001100                  MOV     (R2),@#STMP1    ;SAVE EXPECTED U-ADDR FOR ERROR
15134  056502  104015                          ERROR   15              ;MICROBK. TRAP-TO-4 DID NOT OCCUR
15135
15136  056504  105721                  20$:    TSTB    (R1)+           ;INCREMENT TO NEXT TABLE
15137  056506  005722                          TST     (R2)+           ;ENTRIES AND
15138  056510  000744                          BR      2$              ;CONTINUE
15139
15140  056512  076600                  50$:    MED                     ;GET WHAMI INTO R0
15141  056514  000022                          RDWHAMI
15142  056516  042700  001000                  BIC     #BIT9,R0        ; CLEAR BIT 9
15143  056522  076600                          MED                     ;CLEAR THE FLAG REG. TO
15144  056524  000344                  13$:    WRFLAG                  ;    DISABLE MICROBK. TRAPPING
15145  056526  076600                          MED                     ;CLEAR THE WHAMI REG. TO
15146  056530  000222                  14$:    WRWHAMI                 ;    DISABLE MICROBK. TRAP-TO-4
15147  056532  012737  056260  001010          MOV     #MEDT11,@#$LPERR        ;RESET LOOP ON ERROR POINTER
15148  056540  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE NORMAL ERROR ROUTINE
```

```
15149  056546  012737  000304  177770          MOV     #304,@#UBREAK   ;GENERATE SYNC PULSE ON MED INSTR
15150
15151
15152                                  ;;*****************************************************************
15153                                  ;*TEST 760      PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING
15154                                  ;*      THIS TEST CHECKS THAT THE PROPER PHYSICAL ADDRESS BITS
15155                                  ;*      <17:00> ARE LOGGED UPON ERROR.  THE ERROR IS CAUSED BY
15156                                  ;*      FORCING AN ODD ADDRESS TRAP.  THE ERROR LOG MODE USED
15157                                  ;*      IS "LOG FIRST".  ALSO, THE ODD ADDRESS ERROR BITS IN
15158                                  ;*      THE LOG JAM AND CPU ERROR REGISTER ARE CHECKED.
15159                                  ;;*****************************************************************
15160  056554                          TST760:
15161  056554  012700  000757                 MOV     #757,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15162  056560  000004                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15163  056562  012737  056622  000004  1$:     MOV     #2$,@#4         ;SETUP PC FOR ODD ADDR SERVICE
15164  056570  012737  000340  000006          MOV     #340,@#6
15165  056576  012700  100001                 MOV     #BIT15+BIT0,R0  ;SETUP "LOG FIRST" MODE
15166  056602  076600                 MED
15167  056604  000222                 WRWHAMI
15168  056606  012702  056563                 MOV     #1$+1,R2        ;SAVE ADDRESS OF ODD ADDR. INSTRUCTION
15169  056612  005767  177745                 TST     1$+1            ;DO ODD ADDRESS INSTRUCTION TO FORCE
15170                                                                  ;A JAMUPP & TRAP TO 4
15171  056616  104023                 ERROR   23                      ;*** ODD ADDR. TRAP DID NOT OCCUR
15172  056620  000441                 BR      10$                     ;EXIT TEST
15173  056622  022626          2$:     CMP     (SP)+,(SP)+             ;RESTORE STACK
15174  056624  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW
15175  056632  076600                 MED
15176  056634  000100                 ROLJAM
15177  056636  013701  177766                 MOV     @#CPUERR,R1
15178  056642  032701  000100                 BIT     #BIT6,R1          ;WAS ODD ADDR. ERROR RECORDED BY
15179                                                                  ;THE CPU ERROR REGISTER?
15180  056646  001001                 BNE     3$                      ;BRANCH IF YES
15181  056650  104024                 ERROR   24                      ;*** CPU ERROR REG. DID NOT
15182                                                                  ;REPORT ODD ADDRESS ERROR
15183                                                                  ;READ THE LOG JAM REGISTER
15184  056652  032700  100004          3$:     BIT     #BIT15+BIT2,R0  ;WAS ODD ADDR. ERROR LOGGED BY LOG JAM
15185  056656  001001                 BNE     4$                      ;BRANCH IF YES
15186  056660  104024                 ERROR   24                      ;*** LOG JAM REG.  DID NOT LOG
15187                                                                  ;ODD ADDRESS ERROR CORRECTLY
15188
15189  056662  005005          4$:     CLR     R5                      ;CLR ERROR FLAG
15190  056664  076600                 MED                             ;READ THE LOG PBA REGISTER
15191  056666  000102                 ROLPBA
15192  056670  010003                 MOV     R0,R3                   ;SAVE RECEIVED PHYS ADDR <15:0>
15193  056672  020002                 CMP     R0,R2                   ;WERE BITS <15:00> OF THE PHYSICAL
15194                                                                  ;BUS ADDR. LOGGED CORRECTLY?
15195  056674  001401                 BEQ     5$                      ;BRANCH IF YES
15196  056676  005205                 INC     R5                      ;SET ERROR FLAG
15197  056700  076600          5$:     MED                             ;READ THE LOG SERVICE REGISTER
15198  056702  000101                 ROLSERVICE
15199  056704  000300                 SWAB    R0                      ;GET "PBA 17&16" DOWN TO BIT POSITION 0&1
15200  056706  042700  177774                 BIC     #177774,R0
15201  056712  001002                 BNE     11$                     ;BR IF PHYS ADDR BITS <17:16> LOGGED CORRECTLY
15202  056714  005705                 TST     R5                      ;PREVIOUS ERROR?
15203  056716  001402                 BEQ     10$                     ;BR IF NOT
15204  056720  005001          11$:    CLR     R1                      ;SET UP EXPECTED PA<17:16>
```

```
15205  056722  104026                          ERROR   26              ;*** PHYSICAL BUS ADDR. (17:00)
15206                                                                   ;NOT LOGGED CORRECTLY WHEN
15207                                                                   ;ODD ADDRESS TRAP OCCURRED
15208  056724  005000                  10$:     CLR     R0
15209  056726  076600                          MED                     ;DISABLE "LOG FIRST" MODE
15210  056730  000222                          WRWHAMI
15211
15212
15213                                  ;;*****************************************************************
15214                                  ;*TEST 761      CHECK DISABLE PARITY ERROR TRAP
15215                                          ;*THIS TEST CHECKS THAT PARITY ERROR TRAPS TO LOCATION 114
15216                                          ;*ARE DISABLED WHEN BIT0 OF THE CACHE CONTROL REGISTER IS
15217                                          ;*SET (=1).  A TRAP TO 114 SHOULD NOT OCCUR AND ERROR
15218                                          ;*INFORMATION SHOULD NOT BE LOGGED IN THE LOG PBA, LOG
15219                                          ;*CACHE DATA, OR LOG TAG DATA REGISTERS.  WRONG PARITY IS
15220                                          ;*WRITTEN INTO A TEST LOCATION TO CAUSE THE PARITY ERROR
15221                                          ;*NEEDED IN THIS TEST.
15222                                  ;;*****************************************************************
15223  056732                          TST761:
15224  056732  012700  000760                  MOV     #760,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15225  056736  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15226
15227  056740  012701  064046                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15228  056744  005711                          TST     (R1)            ;MAKE IT A HIT
15229  056746  012737  000100  177746          MOV     #WMP,@#CCR      ;SET WRITE WRONG PARITY BIT
15230  056754  012711  125252                  MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15231  056760  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE PARITY ERROR TRAPS
15232                                                                   ;AND CLEAR WMP
15233  056766  012737  057026  000114          MOV     #1$,@#114       ;SETUP PARITY ERROR VECTOR
15234  056774  012737  000340  000116          MOV     #340,@#116
15235  057002  005000                          CLR     R0
15236  057004  076600                          MED                     ;CLEAR LOG PBA REGISTER
15237  057006  000302                          WRLPBA
15238  057010  076600                          MED                     ;CLEAR LOG CACHE DATA REGISTER
15239  057012  000306                          WRLDATA
15240  057014  076600                          MED                     ;CLEAR LOG CACHE TAG REGISTER
15241  057016  000307                          WRLTAG
15242  057020  005767  005022                  TST     TLOC1           ;READ TEST LOC0 TO FORCE PARITY ERROR
15243  057024  000406                          BR      2$              ;BRANCH IF NO TRAP OCCURS
15244  057026  012700  000200          1$:     MOV     #200,R0
15245  057032  076600                          MED                     ;CLEAN UP THE CACHE
15246  057034  000352                          352                     ;INITIALIZATION CODE
15247  057036  022626                          CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15248  057040  104030                          ERROR   30              ;*** PARITY TRAP TO 114 OCCURRED
15249                                                                   ;WHEN IT SHOULD HAVE BEEN DISABLED
15250  057042  012700  000200          2$:     MOV     #200,R0
15251  057046  076600                          MED                     ;CLEAN UP THE CACHE
15252  057050  000352                          352                     ;INITIALIZATION CODE
15253  057052  012711  125252                  MOV     #125252,(R1)    ;WRITE BAK GOOD PARITY IN TST LOC
15254  057056  012737  000116  000114          MOV     #116,@#114      ;RESTORE ORIGINAL PARITY HANDLER & PSW
15255  057064  005037  000116                  CLR     @#116
15256  057070  005005                          CLR     R5              ;CLEAR ERROR FLAG
15257  057072  076600                          MED                     ;READ LOG PBA REGISTER
15258  057074  000102                          RDLPBA
15259  057076  010003                          MOV     R0,R3           ;SAVE COPY
15260                                                                   ;LOG PBA REG. STILL CLEAR?
```

```
15261  057100  001401                        BEQ     3$              ;BRANCH IF YES
15262  057102  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15263  057104  076600                3$:     MED                     ;READ LOG CACHE DATA REG.
15264  057106  000106                        RDLDATA
15265  057110  010001                        MOV     R0,R1           ;SAVE COPY
15266                                                                 ;LOG CACHE DATA REG. STILL CLEAR?
15267  057112  001401                        BEQ     4$              ;BRANCH IF YES
15268  057114  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15269  057116  076600                4$:     MED                     ;READ LOG CACHE TAG REG.
15270  057120  000107                        RDLTAG
15271  057122  010002                        MOV     R0,R2           ;SAVE COPY
15272                                                                 ;LOG CACHE TAG REG. STILL CLEAR?
15273  057124  001401                        BEQ     5$              ;BRANCH IF YES
15274  057126  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15275  057130  005705                5$:     TST     R5              ;WERE ANY OF LOG REGISTERS CHANGED
15276  057132  001401                        BEQ     6$              ;BRANCH IF NO
15277  057134  104027                        ERROR   27              ;*** ONE OF LOG REGISTERS CHANGED
15278                                                                 ;WHEN ERROR SHOULD NOT HAVE BEEN LOGGED
15279                                                                 ;LOG PBA, LOG DATA & LOG TAG
15280                                                                 ;REGISTER SHOULD BE CLEAR.
15281  057136  005037  177746        6$:     CLR     @#CCR           ;ENABLE PARITY ERROR TRAPS
15282
15283                                ;;*****************************************************************
15284                                ;*TEST 762      CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
15285
15286                                ;*THIS TEST CHECKS THAT ALL OF THE PARITY ERROR BITS (5,6,7)
15287                                ;*OF THE MEMORY ERROR REGISTER ARE SET TO "1" WHEN A CACHE
15288                                ;*PARITY ERROR OCCURS IN THE BACKUP MODE.
15289                                ;;*****************************************************************
15290  057142                        TST762:
15291  057142  012700  000761                MOV     #761,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15292  057146  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15293  057150  012701  064046                MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15294  057154  005711                        TST     (R1)            ;MAKE IT A HIT
15295  057156  012737  000100  177746        MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15296  057164  012711  125252                MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15297  057170  042737  000100  177746        BIC     #WWP,@#CCR      ;CLEAR WWP
15298  057176  012737  057232  000114        MOV     #1$,@#114       ;SETUP NEW TEST HANDLER AT PARITY VECTOR
15299  057204  012737  000340  000116        MOV     #340,@#116
15300  057212  005737  064046                TST     @#TLOC1         ;READ TEST LOC. TO FORCE PARITY ERROR
15301  057216  012700  000200                MOV     #200,R0
15302  057222  076600                        MED                     ;CLEAN UP THE CACHE
15303  057224  000352                        352                     ;INITIALIZATION CODE
15304  057226  104031                        ERROR   31              ;*** PARITY ERROR DID NOT CAUSE TRAP
15305  057230  000405                        BR      2$              ;BRANCH TO 2$
15306  057232  012700  000200        1$:     MOV     #200,R0
15307  057236  076600                        MED                     ;CLEAN UP THE CACHE
15308  057240  000352                        352                     ;INITIALIZATION CODE
15309  057242  022626                        CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15310  057244  022737  000340  177744  2$:     CMP     #000340,@#MEMERR  ;WERE PARITY ERROR BITS (5,6,7) SET
15311                                                                 ;AND CPU ABORT BIT (15) LEFT CLEAR
15312                                                                 ;IN MEMORY ERROR REGISTER?
15313  057252  001403                        BEQ     3$              ;BRANCH IF YES
15314  057254  013700  177744                MOV     @#MEMERR,R0
15315  057260  104032                        ERROR   32              ;*** MEMORY ERROR REGISTER BITS
15316                                                                 ;WERE SET INCORRECTLY
```

```
15317  057262  012737  000116  000114  3$:     MOV     #116,@#114      ;RESTORE OLD PARITY HANDLER PC & PSW
15318  057270  005037  000116                  CLR     @#116
15319
15320                                  ;;*****************************************************************
15321                                  ;*TEST 763      CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
15322
15323                                          ;*THIS TEST CHECKS THAT THE "UNIBUS TIMEOUT" BIT (BIT4)
15324                                          ;*GETS SET IN THE CPU ERROR REGISTER WHEN A TIMEOUT OCCURS.
15325                                          ;*A TIMEOUT TRAP IS FORCED BY REFERENCING BUS ADDRESS 760000.
15326                                          ;*THEN AN ODD ADDRESS ERROR IS FORCED AND IT
15327                                          ;*IS CHECKED IF ONLY BIT (6)-ODD ADDRESS ERROR IS SET
15328                                          ;*(IN CPUERR).  THIS CHECKS THAT THE ERROR LOG IS
15329                                          ;*CONTINUOSLY UPDATED IN THE "LOG CONTINUOUS" MODE.
15330                                  ;;*****************************************************************
15331  057274                          ;ST763:
15332  057274  012700  000762                  MOV     #762,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15333  057300  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15334  057302  012737  057324  000004          MOV     #1$,@#4         ;SETUP NEW PC & PSW FOR THE
15335  057310  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT SERVICE ROUTINE
15336  057316  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT TRAP TO 4 BY
15337                                                                  ;REFERENCING NON-EXISTENT ADDRESS
15338  057322  000461                  BR      6$
15339  057324  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15340  057326  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW FOR TIMEOUT
15341  057334  076600                  MED
15342  057336  000100                  RDLJAM
15343  057340  013701  177766                  MOV     @#CPUERR,R1     ;SAVE CPU ERR REG
15344  057344  022701  000020                  CMP     #BIT4,R1        ;DID "UNIBUS TIMEOUT" BIT IN CPU ERROR
15345                                                                  ;REGISTER GET SET?
15346  057350  001401                  BEQ     2$                      ;BRANCH IF YES
15347  057352  104033                  ERROR   33                      ;*** "UNIBUS TIMEOUT" BIT (BIT4) IN CPU
15348                                                                  ;ERROR REG. DID NOT SET WHEN A
15349                                                                  ;TIMEOUT WAS FORCED
15350                                                                  ;READ THE LOG JAM REGISTER
15351  057354  022700  021200          2$:     CMP     #BIT13+BIT9+BIT7,R0     ;DID "UNIBUS TIMEOUT" BIT (BIT7) SET?
15352                                                                  ;BIT 9= POWER STATUS, ALWAYS SET
15353  057360  001401                  BEQ     3$                      ;BRANCH IF YES
15354  057362  104033                  ERROR   33                      ;*** "UNIBUS TIMEOUT" BIT (BIT7)
15355                                                                  ;DID NOT SET IN LOG JAM REGISTER
15356                                                                  ;WHEN UNIBUS TIMEOUT WAS FORCED
15357  057364  076600                  3$:     MED                     ;READ LOG PBA
15358  057366  000102                  RDLPBA
15359  057370  020027  160000                  CMP     R0,#160000      ;WAS PHYS BA LOGGED CORRECTLY?
15360  057374  001403                  BEQ     5$
15361  057376  012701  160000                  MOV     #160000,R1
15362  057402  104020                  ERROR   20                      ;PHYSICAL BUS ADDRESS WAS
15363                                                                  ;LOGGED WRONG ON A UNIBUS
15364                                                                  ;TIMEOUT
15365  057404  012737  057426  000004  5$:     MOV     #4$,@#4         ;SET UP PC,PSW FOR ODD ADDRESS
15366  057412  012737  000340  000006          MOV     #340,@#6
15367  057420  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS ERROR
15368  057424  000420                  BR      6$
15369  057426  022626                  4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15370  057430  012737  061220  000004          MOV     #BERR,@#4
15371  057436  076600                  MED
15372  057440  000100                  RDLJAM
```

```
15373  057442  013701  177766          MOV     @#CPUERR,R1
15374  057446  022701  000100          CMP     #BIT6,R1        ;ODD ADDR. BUT SET 3
15375  057452  001401                  BEQ     7$
15376  057454  104024                  ERROR   24              ;ODD ADDRESS BIT WAS
15377                                                          ;NOT SET IN THE CPU
15378                                                          ;ERROR REGISTER.  IN LOG
15379                                                          ;CONTINUOUS MODE THE
15380                                                          ;LATEST ERROR SHOULD
15381                                                          ;BE LOGGED
15382  057456  032700  000004  7$:     BIT     #BIT2,R0        ;ODD ADR. BIT SET IN
15383  057462  001001                  BNE     6$              ;LOG JAM?
15384  057464  104024                  ERROR   24              ;ODD ADDRESS BIT WAS
15385                                                          ;NOT SET IN THE LOG
15386                                                          ;JAM REGISTER ON A
15387                                                          ;ODD ADDRESS ERROR
15388  057466  076600          6$:     MED                     ;CHECK IF LAST INTERRUPT VECTOR
15389  057470  000104                  RDLFGINT                ;WAS LOGGED?
15390  057472  120027  000004          CMPB    R0,#4
15391  057476  001401                  BEQ     8$
15392  057500  104036                  ERROR   36              ;LAST ERROR VECTOPR WS NOT LOGGED
15393
15394  057502          8$:
15395
15396
15397                   ;;*****************************************************************
15398                   ;*TEST 764       CHECK ILLEGAL INTERNAL ADDRESS TRAP
15399
15400                   ;*THIS TEST CHECKS THAT A TRAP OCCURS UPON REFERENCING AN
15401                   ;*ILLEGAL INTERNAL ADDRESS AND THAT "ILLEGAL INTERNAL ADDRESS"
15402                   ;*BIT (BIT0) OF THE CPU ERROR REGISTER AND BITS OF LOG JAM
15403                   ;*REG.STER GET SET.  IT ALSO CHECKS IF THE INTERRUPT VECTOR
15404                   ;*(4) IS SAVED AS THE "LAST INTERRUPT VECTOR" IN THE LOG
15405                   ;*FLAG/INTERRUPT REG.
15406                   ;;*****************************************************************
15407  057502          TST764:
15408  057502  012700  000763          MOV     #763,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15409  057506  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15410  057510  012737  057540  000004  MOV     #1$,@#4         ;SETUP NEW HANDLER PC & PSW
15411  057516  012737  000340  000006  MOV     #340,@#6
15412  057524  005037  177746          CLR     @#CCR
15413  057530  012707  177746          MOV     #CCR,PC         ;ILLEGAL INTERNAL ADDRESS TRAP SHOULD OCCUR
15414  057534  104034                  ERROR   34              ;*** ILLEGAL INTERNAL ADDRESS
15415                                                          ;DID NOT RESULT IN A TRAP
15416  057536  000420                  BR      3$              ;BRANCH TO EXIT IF NO TRAP
15417  057540  022626          1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15418  057542  012737  061220  000004  MOV     #BERR,@#4       ;RESTORE OLD HANDLER PC & PSW
15419  057550  076600                  MED
15420  057552  000100                  RDLJAM
15421  057554  013701  177766          MOV     @#CPUERR,R1
15422  057560  032701  000001          BIT     #BIT0,R1        ;DID "ILLEGAL INTERNAL ADDRESS" BIT (0)
15423                                                          ;IN CPU ERROR REGISTER GET SET?
15424  057564  001001                  BNE     2$              ;BRANCH IF YES
15425  057566  104035                  ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS
15426                                                          ;BIT DID NOT SET IN CPU ERROR REG.
15427                                                          ;READ THE LOG JAM REG.
15428  057570  032700  000040  2$:     BIT     #BIT5,R0        ;DID "ILLEGAL INTERNAL ADDRESS" BIT (5)
```

```
15429
15430  057574  001001                          BNE     3$              ;IN LOG JAM REG. GET SET
15431  057576  104035                          ERROR   35              ;BRANCH IF YES
15432                                                                   ;*** ILLEGAL INTERNAL ADDRESS BIT
15433  057600                          3$:                             ;DID NOT SET IN LOG JAM REG.
15434
15435
15436                                  ;;*****************************************************************
15437                                  ;*TEST 765      CHECK LOG SERVICE & MEMERR LOGS LO-BYTE & TAG, IN CACHE ABORT MODE
15438                                          ;TEST CHECKS THAT "LO BYTE PARITY" "HI BYTE PARITY" AND "TAG PARITY"
15439                                          ;BITS CAN SET IN "LOG SERVICE" REGISTERS.  IT IS ALSO
15440                                          ;CHECKED THAT THE PROPER TAG AND DATA BITS GET STORED
15441                                          ;IN THE "LOG CACHE DATA, "LOG CACHE TAG/CPU" AND THE
15442                                          ;"MEMORY ADDRESS REGISTER" WHEN A PARITY ERROR IS
15443                                          ;FORCED.
15444                                          ;IT IS CHECKED IF THE INSTRUCTION WAS ABORTED AND THE
15445                                          ;LOG FLAG/INTERRUPT REGISTER LOGGED THE LAST INTERRUPT
15446                                          ;VECTOR.
15447                                  ;;*****************************************************************
15448  057600                          $ST765:
15449  057600  012700  000764                 MOV     #764,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15450  057604  000004                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15451
15452  057606  012737  000201  177746         MOV     #DPTRP+PABORT,@#CCR  ;DISABLE PARITY TRAPS (CACHE)
15453  057614  005037  001062                 CLR     @#$REGO
15454  057620  012701  064046                 MOV     #TLOC1,R1       ;GET POINTER TO TEST LOC.
15455  057624  005711                         TST     (R1)            ;MAKE IT A HIT
15456  057626  052737  000100  177746         BIS     #WWP,@#CCR      ;WRITE WRONG PARITY SET
15457  057634  012711  125252                 MOV     #125252,(R1)    ;WRITE TEST LOCATION WITH WRONG PARITY
15458  057640  042737  000100  177746         BIC     #WWP,@#CCR      ;CLEAR WWP
15459  057646  012700  100001                 MOV     #BIT15+BIT0,R0
15460  057652  076600                         MED                     ;ENABLE "LOG FIRST" MODE, AND
15461  057654  000222                         WRWHAMI                 ;ERROR LOGGING
15462  057656  042737  000001  177746         BIC     #DPTRP,@#CCR    ;ENABLE CACHE PARITY TRAPS
15463  057664  012737  057712  000114         MOV     #PTRP1,@#114    ;NEW PARITY TRAP SERVICE
15464  057672  016737  004150  001062         MOV     TLOC1,@#$REGO   ;READ TEST LOC, FORCE PARITY ERROR
15465  057700  012700  000200                 MOV     #200,R0
15466  057704  076600                         MED                     ;CLEAN UP THE CACHE
15467  057706  000352                         352                     ;INITIALIZATION CODE
15468  057710  104031                         ERROR   31              ;*** CACHE PARITY ERROR TRAP
15469                                                                   ;DID NOT OCCUR WHEN
15470                                                                   ;TEST LOC WITH BAD PARITY
15471                                                                   ;WAS READ
15472
15473                                                                   ;ENTER HERE IF PARITY TRAP OCCURRED
15474  057712  012700  000200          PTRP1: MOV     #200,R0
15475  057716  076600                         MED                     ;CLEAN UP THE CACHE
15476  057720  000352                         352                     ;INITIALIZATION CODE
15477  057722  012737  000001  177746         MOV     #DPTRP,@#CCR    ;DISABLE CACHE PARITY ERROR TRAPS
15478  057730  012737  000116  000114         MOV     #116,@#114      ;REESTABLISH OLD SERVICE VECTORS
15479  057736  005037  000116                 CLR     @#116
15480  057742  022626                         CMP     (SP)+,(SP)+
15481  057744  005737  001062                 TST     @#$REGO         ;WAS THE INSTRUCTION ABORTED ON
15482                                                                   ;CACHE PARITY ERROR (ABORT MODE)?
15483  057750  001401                         BEQ     1$              ;YES
15484  057752  104041                         ERROR   41              ;INSTRUCTION HAVING CACHE PARITY
```

```
15485                                                   ;ERROR WAS NOT ABORTED, IN THE
15486                                                   ;CACHE ABORT MODE.
15487  057754  076600                  1$:     MFD       ;READ THE "LOG SERVICE" REGISTER
15488  057756  000101                          RDLSERVICE
15489  057760  010004                          MOV       R0,R4           ;COPY
15490  057762  042704  177435                BIC       #^C<LO+HI+TAG+BIT1>,R4  ;MASK ALL BUT LO,HI,TAG BITS
15491  057766  022704  000342                CMP       #342,R4         ;LO,HI ,TAG, CACHE PARITY BITS SET? IN "SERVICE"
15492  057772  001401                        BEQ       2$              ;YES
15493  057774  104042                        ERROR     42              ;*** "LO BYTE" PARITY ERROR
15494                                                   ;AND "TAG" PARITY ERROR BITS
15495                                                   ;WERE NOT LOGGED CORRECTLY IN "LOG
15496                                                   ;SERVICE" REGISTER, WHEN PARITY
15497                                                   ;ERROR TRAP WAS FORCED.
15498                                                   ;CLEAR BITS ARE ACTIVE.
15499  057776  013700  177744          2$:     MOV       @#MEMERR,R0     ;GET MEM ERR REG
15500  060002  022700  100340                CMP       #HI+LO+TAG+BIT15,R0  ;DID "LO BYTE" "HI BYTE" AND "TAG"
15501                                                   ;PARITY ERROR BITS SET IN
15502                                                   ;THE MEMORY ERROR REGISTER?
15503  060006  001401                        BEQ       3$              ;YES
15504  060010  104043                        ERROR     43              ;*** "LO BYTE" "HI BYTE" AND "TAG" PARITY
15505                                                   ;ERROR BITS DID NOT SET
15506                                                   ;CORRECTLY IN THE MEMORY
15507                                                   ;ERROR REGISTER
15508  060012  076600                  3$:     MFD       ;READ "LOG PBA" REGISTER
15509  060014  000102                          RDLPBA
15510  060016  020027  064046                CMP       R0,#TLOC1       ;DID "LOG PBA" CONTAIN CORRECT
15511                                                   ;PHYSICAL BUS ADDRESS-WHERE
15512                                                   ;THE PARITY ERROR OCCURRED?
15513  060022  001403                        BEQ       4$              ;YES
15514  060024  012701  064046                MOV       #TLOC1,R1       ;EXPECTED PBA
15515  060030  104020                        ERROR     20              ;*** PHYSICAL BUS ADDRESS
15516                                                   ;(WHERE PARITY ERROR OCCURRED)
15517                                                   ;WAS NOT LOGGED CORRECTLY
15518                                                   ;WHEN CACHE PARITY ERROR WAS FORCED
15519  060032  076600                  4$:     MFD
15520  060034  000107                          RDLTAG                    ;READ "LOG CACHE TAG" REGISTER
15521  060036  000300                          SWAB      R0
15522  060040  012701  064046                MOV       #TLOC1,R1       ;SHIFT RIGHT (3 TIMES) THE 16 BIT
15523  060044  000301                          SWAB      R1
15524  060046  106201                          ASRB      R1              ;PHYSICAL BUS ADDRESS OF THE
15525  060050  106201                          ASRB      R1              ;TEST LOCATION
15526  060052  106201                          ASRB      R1
15527  060054  052701  000200                BIS       #BIT7,R1        ;FUDGE TAGE BIT
15528  060060  120100                        CMPB      R1,R0           ;WAS THE CORRECT TAG LOGGED?
15529  060062  001401                        BEQ       5$              ;YES
15530  060064  104017                        ERROR     17              ;TAG BITS WERE NOT LOGGED
15531                                                   ;CORRECTLY, WHEN CACHE
15532                                                   ;PARITY ERROR WAS FORCED
15533  060066  076600                  5$:     MFD       ;READ CACHE DATA
15534  060070  000106                          RDLDATA
15535  060072  020027  125252                CMP       R0,#125252      ;CACHE DATA LOGGED CORRECTLY?
15536  060076  001403                        BEQ       6$
15537  060100  012701  125252                MOV       #125252,R1      ;EXPECTED DATA
15538  060104  104016                        ERROR     16
15539
15540  060106  012700  000001          6$:     MOV       #BIT0,R0        ;SET UP LOG CONTINUOUS
```

```
15541  060112  076600                          MED
15542  060114  000222                          WRWHAMI
15543  060116  012737  060130  000004          MOV     #7$,@#4         ;SETUP CPU VECTOR
15544  060124  005737  160000                  TST     @#160000        ;FORCE TIMEOUT & TRAP TO 7$
15545  060130  022626                  7$:     CMP     (SP)+,(SP)+
15546  060132  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE CPU VECTOR
15547  060140  076600                          MED                     ;READ LOG FLAG/INTERRUPT REGISTER
15548  060142  000104                          RDLFGINT
15549  060144  120027  000114                  CMPB    R0,#114         ;DID LO BYTE CONTAIN VECTOR 114?
15550  060150  001403                          BEQ     8$
15551  060152  010037  001062                  MOV     R0,@#SREG0
15552  060156  104036                          ERROR   36              ;LAST INTERRUPT VECTOR WAS NOT
15553                                                                  ;LOGGED CORRECTLY IN FLAG REGISTER
15554                                                                  ;WHEN A CACHE PARITY ERROR WAS
15555                                                                  ;FORCED.
15556  060160                          8$:
15557
15558                                  ;;**************************************************************
15559                                  ;*TEST 766      CHECK "LOG FIRST" MODE OF ERROR LOGGING
15560                                  ;*THIS TEST CHECKS THE "LOG FIRST" MODE OF ERROR LOGGING.
15561                                  ;*THE "LOG FIRST" MODE IS ENABLED.  THEN A TIME-OUT TRAP
15562                                  ;*IS FORCED, BIT 4 OF CPU ERROR REGISTER SHOULD BE SET.
15563                                  ;*THEN AN ODD ADDRESS TRAP IS FORCED.  HOWEVER, THIS
15564                                  ;*TIME THE ERROR SHOULD NOT BE LOGGED; BIT 6 (ODD
15565                                  ;*ADDRESS) SHOULD NOT BE SET BECAUSE THE ERROR LOG
15566                                  ;*IS LOCKED UP AFTER THE FIRST ERROR.
15567
15568                                  ;*THEN, THE ERROR LOG IS ENABLED (BY SETTING BIT 0 OF
15569                                  ;*WRWHAMI).  AN ODD ADDRESS ERROR IS FORCED AGAIN AND IT IS
15570                                  ;*CHECKED THAT THIS TIME THE ERROR IS LOGGED, (BIT 6-ODD
15571                                  ;*ADDRESS SHOULD BE SET IN CPU ERROR REGISTER).
15572                                  ;;**************************************************************
15573  060160                          TST766:
15574  060160  012700  000765                  MOV     #765,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15575  060164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15576
15577  060166  012700  100001                  MOV     #BIT15+BIT0,R0  ;SET UP "LOG FIRST MODE
15578  060172  076600                          MED
15579  060174  000222                          WRWHAMI
15580  060176  012737  060220  000004          MOV     #1$,@#4         ;SET UP NEW PC & PSW FOR
15581  060204  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT
15582  060212  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT
15583  060216  000462                          BR      5$              ;SKIP TEST IF NO TIMEOUT
15584
15585  060220  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15586                                                                  ;BIT 4 OF CPU ERROR REGISTER
15587                                                                  ;SHOULD HAVE SET
15588  060222  012737  060236  000004          MOV     #2$,@#4         ;SET UP NEW PC FOR ODD ADDRESS
15589  060230  005767  177765                  TST     1$+1            ;FORCE ODD ADDRESS TRAP
15590  060234  000453                          BR      5$              ;SKIP TEST IF NO ODD ADDRESS TRAP
15591
15592  060236  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15593  060240  012737  061220  000004          MOV     #BERR,@#4
15594  060246  076600                          MED
15595  060250  000100                          RDLJAM
15596  060252  013701  177766                  MOV     @#CPUERR,R1
```

```
15597  060256  022701  000020            CMP     #BIT4,R1        ;"TIMEOUT" BIT SHOULD BE STILL
15598                                                             ;SET, CHECK
15599  060262  001402                    BEQ     3$
15600  060264  104033                    ERROR   33              ;*** SECOND ERROR (ODD ADDRESS)
15601                                                             ;UPDATED THE ERROR LOG IN
15602                                                             ;THE LOG FIRST MODE.  BIT 4
15603                                                             ;(UNIBUS TIMEOUT) SHOULD BE
15604                                                             ;STILL SET FROM THE FIRST
15605                                                             ;ERROR
15606  060266  000436                    BR      5$              ;SKIP THE REST
15607  060270  032700  100004    3$:     BIT     #BIT2+BIT15,R0  ;CHECK THAT ODD ADRES ERROR BITS NOT
15608  060274  001401                    BEQ     6$              ;SET IN LOG JAM.  NOTE LOG FIRST
15609                                                             ;MODE SHOULD INHIBIT FURTHER
15610                                                             ;ERROR LOGGING
15611  060276  104037                    ERROR   37              ;ODD ADDRESS ERROR BITS GOT SET IN LOG JAM
15612                                                             ;THEY SHOULD NOT BE SINCE LOG FIRST MODE
15613                                                             ;INHIBITS ERROR LOGGING AFTER THE FIRST ERROR
15614  060300  012700  100001    6$:     MOV     #BIT15+BIT0,R0  ;ENABLE ERROR LOG AGAIN IN
15615                                                             ;LOG FIRST MODE
15616  060304  076600                    MED
15617  060306  000222                    WRWHAMI
15618  060310  012737  060332  000004    MOV     #4$,@#4         ;SET UP NEW PC & PSW FOR
15619  060316  012737  000340  000006    MOV     #340,@#6        ;ODD ADDRESS ERROR
15620  060324  005767  177741            TST     3$+1            ;FORCE ODD ADDRESS TRAP
15621  060330  000415                    BR      5$              ;SKIP IF NO TRAP
15622  060332  022626            4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15623                                                             ;RESTORE OLD PC(4), PSW(6)
15624  060334  012737  061220  000004    MOV     #BERR,@#4
15625  060342  022737  000100  177766    CMP     #BIT6,@#CPUERR  ;THE ERROR LOG FROM PREVIOUS
15626                                                             ;ERROR SHOULD BE OVER WRITTEN.
15627                                                             ;ODD ADDRESS BIT SHOULD
15628                                                             ;BE SET, BECAUSE THE ERROR
15629  060350  001405                    BEQ     5$              ;LOG WAS ENABLED.
15630                                                             ;OK, IF YES
15631  060352  076600                    MED
15632  060354  000100                    RDLJAM
15633  060356  013701  177766            MOV     @#CPUERR,R1
15634  060362  104040                    ERROR   40              ;THE ERROR LOG WAS NOT UPDATED
15635                                                             ;(UPON AN ODD ADDRESS ERROR)
15636                                                             ;AFTER THE LOG WAS ENABLED.
15637                                                             ;AT THIS FORMAT BIT 6 OF
15638                                                             ;CPU ERROR REGISTER SHOULD
15639                                                             ;BE SET.  IT WAS NOT.
15640  060364  012737  061220  000004    5$:     MOV     #BERR,@#4       ;RESTORE OLD PC(4), PSW(6)
15641  060372  012700  000001            MOV     #BIT0,R0
15642  060376  076600                    MED                     ;PUT THE LOGGING BACK INTO
15643  060400  000222                    WRWHAMI                 ;"CONTINUOUS" MODE
15644
15645                              ;;*****************************************************************
15646                              ;*TEST 767      CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
15647                              ;;*****************************************************************
15648  060402                      TST767:
15649  060402  012700  000766            MOV     #766,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15650  060406  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15651
15652  060410  012737  060420  000030    MOV     #1$,@#30        ;LOAD EMT VECTOR WITH 1$
```

```
15653  060416  104000                          EMT                     ;FIRST INTERRUPT -- EMT
15654  060420  022626                  1$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15655  060422  012737  061620  000030          MOV     #$ERROR,@#30    ;RESTORE VECTOR
15656  060430  012737  060442  000004          MOV     #2$,@#4         ;SET UP CPU VECTOR
15657  060436  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15658  060442  022626                  2$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15659  060444  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15660  060452  076600                          MED                     ;CHECK FLAG
15661  060454  000104                          ROLFGINT
15662  060456  120027  000030                  CMPB    R0,#30          ;EMT VECTOR LAST LOGGED?
15663  060462  001401                          BEQ     3$              ;BR IF YES
15664  060464  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15665                                                                  ;LO BYTE OF LOG FLAG/INT REG S/B=30
15666
15667  060466  012737  060476  000020  3$:     MOV     #4$,@#20        ;LOAD IOT VECTOR WITH 4$
15668  060474  000004                          IOT                     ;SECOND INTERRUPT-SHOULD LOAD LOG FLAG REG
15669  060476  022626                  4$:     CMP     (SP)+,(SP)+     ;CLEANUP STACK
15670  060500  012737  061260  000020          MOV     #$SCOPE,@#20    ;RESTORE IOT VECTOR
15671  060506  012737  060520  000004          MOV     #5$,@#4         ;SET UP CPU VECTOR
15672  060514  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15673  060520  022626                  5$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15674  060522  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15675  060530  076600                          MED                     ;CHECK FLAG
15676  060532  000104                          ROLFGINT
15677  060534  120027  000020                  CMPB    R0,#20          ;IOT VECTOR LAST LOGGED?
15678  060540  001401                          BEQ     6$              ;BR IF YES
15679  060542  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15680                                                                  ;LOW BYTE S/B = 20
15681
15682  060544  012700  000767          6$:     MOV     #STN-1,R0       ;SET UP FOR MISSED TEST CHECK AND
15683                                                                  ;FULL WORD TEST NUMBER FOR APT
15684
15685                                  .ENABLE AMA
15686
15687
15688                                  .SBTTL  END OF PASS ROUTINE
15689
15690                                  ;;*****************************************************************
15691                                  ;*INCREMENT THE PASS NUMBER ($PASS)
15692                                  ;*IF THERES A MONITOR GO TO IT
15693                                  ;*IF THERE ISN'T JUMP TO INIT
15694
15695  060550                          $EOP:
15696  060550  000004                          SCOPE
15697  060552  005037  001110                  CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
15698  060556  005237  001126                  INC     $PASS           ;;INCREMENT THE PASS NUMBER
15699  060562  042737  100000  001126          BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
15700  060570  005327                          DEC     (PC)+           ;;LOOP?
15701  060572  000001                  $EOPCT: .WORD   1
15702  060574  003027                          BGT     $DOAGN          ;;YES
15703  060576  012737                          MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
15704  060600  000001                  $ENDCT: .WORD   1
15705  060602  060572                          $EOPCT
15706  060604  104401  065103                  TYPE    ,EOP1           ;TYPE "END PASS #"
15707  060610  013746  001126                  MOV     $PASS,-(SP)     ;SAVE $PASS FOR TYPOUT
15708  060614  104402                          TYPOC                   ;TYPE PASS NUMBER IN OCTAL
```

END OF PASS ROUTINE

```
15709  060616  104401  065121                  TYPE    $EOP2                   ;TYPE "ERROR COUNT ="
15710  060622  013746  001012                  MOV     $ERTTL,-(SP)            ;SAVE ERROR TOTAL FOR TYPOUT
15711  060626  104402                          TYPOC                           ;TYPE ERROR TOTAL
15712  060630  104401  001115                  TYPE    $CRLF
15713  060634  013700  000042          $GET42: MOV     @#42,R0                 ;;GET MONITOR ADDRESS
15714  060640  001405                          BEQ     $DOAGN                  ;;BRANCH IF NO MONITOR
15715  060642  000005                          RESET                           ;;CLEAR THE WORLD
15716  060644  004710                  $ENDAD: JSR     PC,(R0)                 ;;GO TO MONITOR
15717  060646  000240                          NOP                             ;;SAVE ROOM
15718  060650  000240                          NOP                             ;;FOR
15719  060652  000240                          NOP                             ;;ACT11
15720  060654                          $DOAGN:
15721  060654  000137                          JMP     @(PC)+                  ;;RETURN
15722  060656  003262                  $RTNAD: .WORD   INIT
15723  060660     377      377      000  $ENULL: .BYTE   -1,-1,0                 ;;NULL CHARACTER STRING
15724           060664                          .EVEN
```

```
15725                                   ;XXXX
```

```
15781
15782                                   ; ************************************************************************
15783                                           .SBTTL "T" BIT SERVICE ROUTINE
15784                                   ; ************************************************************************
15785
15786  061070  062716  000002          TBSER:  ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
15787  061074  042766  000020  000002          BIC     #20,2(SP)       ;TURN OFF THE "T" BIT
15788  061102  000006                          RTT                     ;RETURN TO THE CALLING TEST
15789
15790                                   .SBTTL  MICROBREAK TRAP SERVICE ROUTINE
15791                                   ;;**************************************************************
15792                                   ;*      THIS ROUTINE MERELY SETS A FLAG
15793                                   ;*      WHEN THE ROUTINE HAS BEEN ENTERED
15794                                   ;*
15795  061104  005237  061112          BKROUT: INC     BKFLAG          ;SET MICROBREAK FLAG TO
15796                                                                   ;INDICATE TRAP TO 4 OCCURRED
15797  061110  000002                          RTI                     ;RETURN FROM TRAP
15798  061112  000000                  BKFLAG: .WORD   0               ;MICROBREAK TRAP FLAG
15799
15800                                   ; ************************************************************************
15801                                           .SBTTL  RSVD INSTRUCTION TRAP SERVICE ROUTINE
15802                                   ; ************************************************************************
15803
15804                                   ;THIS ROUTINE SERVICES UNEXPECTED RESERVED INSTRUCTION TRAP ERRORS
15805                                   ;IT RESULTS IN PRINTING THE ERROR MESSAGE: "TRAPPED TO 10 PC=XXXXXX"
15806                                   ;WHERE XXXXXX IS THE ADDRESS CONTAINING THE INSTRUCTION WORD THAT
15807                                   ;SPRUNG THE TRAP.  AFTER PRINTING THE ERROR MESSAGE AN ATTEMPT IS
15808                                   ;MADE TO RESTART THE PROGRAM AT THE BEGINNING.
15809
15810                                   ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A
15811                                   ;PREVIOUS RSVD INSTRUCTION TRAP OR AN UNEXPECTED BUS ERROR THE PROGRAM
15812                                   ;WILL HALT.  AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE
15813                                   ;TO THE TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15814
15815                                           ;[SP]   PC+2    OF 2ND TRAP
15816                                           ;[SP]+2 PSW
15817                                           ;[SP]+4 PC+2    OF 1ST TRAP
15818                                           ;[SP]+6 PSW
15819
15820                                           ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE FOLLOWING
15821                                           ;INFORMATION:
15822
15823                                                   ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15824                                                                   ;TRAP (PC AT TIME OF ERROR HALT INDICATES
15825                                                                   ;WHICH OCCURRED FIRST)
15826                                                   ;[CATERR]=2     TWO SUCCESSIVE BUS ERROR TRAPS
15827                                                   ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15828
15829                                   ;THE CONTENTS OF R0 AT THE TIME OF THE
15830                                   ;HALT PROVIDES FURTHER INFORMATION AS TO THE LAST TEST BEING EXECUTED
15831                                   ;WHEN THE TRAPS OCCURRED.
15832
15833                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION
15834                                   ;TESTS TO VERIFY THE RSVD INSTR TRAP MECHANISM PRIOR TO ACTIVATING THE SERVICE
15835                                   ;ROUTINE
15836
```

```
 15837  061114  005137  063246          RSVTST: COM     RSVFLG          ;SET RSVD INSTR TRAP TEST FLAG
 15838  061120  000002                          RTI                     ;RETURN TO BASIC TEST
 15839
 15840  061122  005737  063252          RSERR:  TST     @#CATERR        ;ANY PENDING CATASTROPHIC ERRORS
 15841  061126  001025                          BNE     INCRSV          ;BE IF YES
 15842  061130  105237  063253                  INCB    @#1+CATERR      ;SET RSVD INSTR FLAG
 15843  061134  032777  010000  117676          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT ?
 15844  061142  001015                          BNE     RESTAR          ;BR IF YES
 15845  061144  104401                          TYPE                    ;GO TYPE "TRAPPED TO 10 PC="
 15846  061146  065243                          RSMSG
 15847  061150  011646                  RSBERT: MOV     (SP),-(SP)      ;GET ERROR PC ON STACK FOR PRINTING
 15848  061152  104402                          TYPOC                   ;TYPE THE ERROR PC
 15849  061154  104401                          TYPE                    ;OUTPUT CR / LF
 15850  061156  001115                          $CRLF
 15851  061160  005237  001012                  INC     @#SERTTL        ;COUNT THE ERROR
 15852  061164  032777  100000  117646          BIT     #BIT15,@SWR     ;HALT ON ERROR?
 15853  061172  001401                          BEQ     RESTAR          ;BR IF NOT
 15854  061174  000000                          HALT                    ;HALT ON ERROR--PRESS CONTINUE TO RESTART
 15855  061176  000137  003262          RESTAR: JMP     @#INIT          ;GO ATTEMPT RESTART
 15856  061202  105237  063253          INCRSV: INCB    @#1+CATERR      ;INCREMENT RSVD INSTR FLAG
 15857  061206  000000                          HALT                    ;CATASTROPHIC ERROR HALT
 15858  061210  000772                          BR      RESTAR          ;DEPRESSING CONTINUE WILL CAUSE
 15859                                                                  ;ATTEMPT TO RESTART.
 15860
 15861                          ; ***************************************************************
 15862                                  .SBTTL  BUS ERROR TRAP SERVICE ROUTINE
 15863                          ; ***************************************************************
 15864
 15865                          ;THIS ROUTINE SERVICES UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD ADDRESS
 15866                          ;ERRORS, STACK OVERFLOW, AND ILLEGAL INSTRUCTIONS). IT RESULTS IN PRINTING THE
 15867                          ;ERROR MESSAGE: "TRAPPED TO 4 PC =XXXXXX" WHERE XXXXXX IS THE
 15868                          ;CONTENTS OF THE PC WHEN THE TRAP WAS SPRUNG.  AFTER PRINTING THE
 15869                          ;ERROR MESSAGE AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
 15870                          ;THE BEGINNING.
 15871
 15872                          ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A PREVIOUS
 15873                          ;RSVD INSTR TRAP OR A PREVIOUS BUS ERROR, THE PROGRAM WILL HALT.
 15874                          ;AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE TO THE
 15875                          ;TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
 15876
 15877                                  ;[SP]   PC+2    OF 2ND TRAP
 15878                                  ;[SP]+2 PSW
 15879                                  ;[SP]+4 PC+2    OF 1ST TRAP
 15880                                  ;[SP]+6 PSW
 15881
 15882                                  ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE LOLLOWING
 15883                                  ;INFORMATION:
 15884
 15885                                          ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
 15886                                                          ;TRAP (PC AT TIME OF ERROR HALT
 15887                                                          ;INDICATES WHICH OCCURRED FIRST)
 15888                                          ;[CATERR]=2     TWO SUCCESSIVE BUS ERRORS
 15889                                          ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
 15890
 15891                          ;THE CONTENTS OF R0 AT THE TIME OF
 15892                          ;THE HALT PROVIDED FURTHER INFORMATION AS TO THE TEST IN PROGRESS
```

```
15893                                   ;WHEN THE TRAPS OCCURRED.
15894
15895                                   ;THE CONTENTS OF THE SP CAN BE USED TO INDICATE IF STACK OVERFLOW CAUSED
15896                                   ;THE BUSS ERROR TRAP(S) AS SHOWN BELOW:
15897
15898                                           ;400>(SP)>336   YELLOW ZONE
15899                                           ;(SP)=0         RED ZONE
15900
15901                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION TESTS TO
15902                                   ;VERIFY THAT THE BUS ERROR TRAP MECHANISM WORKS PRIOR TO ACTIVATING
15903                                   ;THE SERVICE ROUTINE
15904
15905  061212  005137  063250          BETST:  COM     BERFLG          ;SET BUS ERROR TRAP TEST FLAG
15906  061216  000002                          RTI                     ;RETURN TO BASIC TEST
15907
15908  061220  005737  063252          BERR:   TST     @#CATERR        ;ANY CATASTROPHIC ERRORS PENDING?
15909  061224  001011                          BNE     2$              ;BR IF YES
15910  061226  105237  063252                  INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15911  061232  032777  010000  117600          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT
15912  061240  001356                          BNE     RESTAR          ;BR IF YES
15913  061242  104401                          TYPE                    ;PRINT "TRAP TO 4" MESSAGE
15914  061244  065216                          BEMSG
15915  061246  000740                          BR      RSBERT          ;TYPE REST OF BUS ERROR MESSAGE
15916
15917  061250  105237  063252          2$:     INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15918  061254  000000                          HALT                    ;CATASTROPHIC ERROR HALT-SCHOOLS OUT
15919  061256  000747                          BR      RESTAR          ;DEPRESS CONTINUE TO ATTEMPT RESTART
15920
15921
15922
15923                                   .SBTTL  SCOPE HANDLER ROUTINE
15924
15925                                   ;;*****************************************************************
15926                                   ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
15927                                   ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
15928                                   ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
15929                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
15930                                   ;*SW14=1        LOOP ON TEST
15931                                   ;*SW11=1        INHIBIT ITERATIONS
15932                                   ;*SW09=1        LOOP ON ERROR
15933                                   ;*CALL
15934                                   ;*      SCOPE           ;;SCOPE=IOT
15935
15936  061260                          $SCOPE:
15937  061260  020037  001124                  CMP     R0,@#$TESTN     ;ANY MISSED TESTS ?
15938  061264  001406                          BEQ     10$             ;BR IF NOT
15939  061266  012737  061276  001112          MOV     #12$,@#$ESCAPE  ;NO ERROR LOOPING
15940  061274  104011                          ERROR   11              ;MISSED TESTS ERROR CALL
15941  061276  005037  001112          12$:    CLR     @#$ESCAPE       ;NORMAL ERROR LOOPING
15942  061302  110037  001002          10$:    MOVB    R0,@#$TSTNM     ;INSURE TSTNUM IS CORRECT
15943  061306  032777  002000  117524          BIT     #SW10,@SWR      ;LOOP ON SELECTED TEST?
15944  061314  001411                          BEQ     11$             ;BR IF NO
15945  061316  017737  117516  063242          MOV     @SWR,@#SELTST   ;GET CONTENTS OF SWITCHES
15946  061324  042737  177000  063242          BIC     #177000,@#SELTST        ;MASK OUT SWR<15:9>
15947  061332  020037  063242                  CMP     R0,@#SELTST     ;IS THIS THE SELECTED TEST?
15948  061336  001510                          BEQ     $OVER           ;BR IF YES
```

```
15949  061340                                  11$:
15950  061340  032777  040000  117472  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
15951  061346  001104                          BNE     $OVER           ;;YES IF SW14=1
15952                                          ;#####START OF CODE FOR THE XOR TESTER#####
15953  061350  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE "XOR" TESTER CHANGE
15954                                                                  ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
15955  061352  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
15956  061356  012737  061376  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
15957  061364  005737  177060                  TST     @#177060        ;;TIME OUT ON XOR?
15960  061370  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15959  061374  000453                          BR      $SVLAD          ;;GO TO THE NEXT TEST
15960  061376  022626                  5$:     CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
15961  061400  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15962  061404  000413                          BR      7$              ;;LOOP ON THE PRESENT TEST
15963  061406                                  6$:;#####END OF CODE FOR THE XOR TESTER#####
15964  061406  105737  001003          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
15965  061412  001421                          BEQ     3$              ;;BR IF NO
15966  061414  123737  001015  001003          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
15967  061422  101015                          BHI     3$              ;;BR IF NO
15970  061424  032777  001000  117406          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
15971  061432  001404                          BEQ     4$              ;;BR IF NO
15972  061434  013737  001010  001006  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
15973  061442  000446                          BR      $OVER
15974  061444  105037  001003          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
15975  061450  005037  001110                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
15976  061454  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
15977  061456  032777  004000  117354  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
16000  061464  001011                          BNE     1$              ;;BR IF YES
16001  061466  005737  001126                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
16002  061472  001406                          BEQ     1$              ;;      INHIBIT ITERATIONS
16003  061474  005237  001004                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
16004  061500  023737  001110  001004          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
16005  061506  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
16006  061510  012737  000001  001004  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
16007  061516  013737  061610  001110          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
16010  061524  105237  001002          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
16011  061530  113737  001002  001124          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
16012  061536  011637  001006                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
16013  061542  011637  001010                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
16014  061546  005037  001112                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
16015  061552  112737  000001  001015          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
16016  061560  013777  001002  117254  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
16017  061566  013716  001006                  MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
16020  061572  120037  001002                  CMPB    R0,@#$TSTNM     ;WAS $TSTNM INCREMENTED?
16021  061576  001401                          BEQ     10$             ;BR IF NOT
16022  061600  005200                          INC     R0              ;INCREMENT TEST NUMBER
16023  061602  010037  001124          10$:    MOV     R0,@#$TESTN     ;FIX $TESTN TO BE WORD COUNT, NOT BYTE
16024  061606  000002                          RTI
16025  061610  000200                  $MXCNT: 200                     ;;MAX. NUMBER OF ITERATIONS
16026
16027  061612  005137  063244          SCOPEA: COM     @#SCOFLG        ;THESE TWO INSTRUCTIONS ARE
16030  061616  000002                          RTI                     ;USED IN THE BASIC TESTS TO
16031                                                                  ;VERIFY THE IOT LINKAGE
16032
16033
16034
```

```
16005                                   .SBTTL  ERROR HANDLER ROUTINE
16006
16007                                   ;;**********************************************************************
16008                                   ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
16009                                   ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
16010                                   ;*AND GO TO $ERRTYP ON ERROR
16011                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
16012                                   ;*SW15=1        HALT ON ERROR
16013                                   ;*SW13=1        INHIBIT ERROR TYPEOUTS
16014                                   ;*SW09=1        LOOP ON ERROR
16015                                   ;*CALL
16016                                   ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER
16017
16018  061620                           $ERROR:
16019  061620  010546                           MOV     R5,-(SP)        ;SAVE R5 ON STACK
16020  061622  012705  001060                   MOV     #$REGAD,R5      ;GET POINTER
16021  061626  016625  000004                   MOV     4(SP),(R5)+     ;SAVE ERROR PSW IN $REGAD FOR TYPOUT
16022  061632  010025                           MOV     R0,(R5)+        ;SAVE R0 FOR TYPEOUTS
16023  061634  010125                           MOV     R1,(R5)+        ;SAVE R1 IN $REG1
16024  061636  010225                           MOV     R2,(R5)+        ;SAVE R2 IN $REG2, ETC.
16025  061640  010325                           MOV     R3,(R5)+
16026  061642  010425                           MOV     R4,(R5)+
16027  061644  022715  177777                   CMP     #-1,(R5)        ;IS SP  ALREADY STORED IN $REG5?
16028  061650  001001                           BNE     10$             ;BR IF YES
16029  061652  010615                           MOV     SP,(R5)         ;PUT SP IN $REG5 FOR TYPOUT
16030  061654  012505                   10$:    MOV     (SP)+,R5        ;RESTORE R5
16031  061656  105237  001003           7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
16032  061662  001775                           BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
16033  061664  013777  001002  117150           MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
16034  061672  005237  001012                   INC     $ERTTL          ;;INC THE ERROR COUNT
16035  061676  011637  001016                   MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
16036  061702  162737  000002  001016           SUB     #2,$ERRPC
16037  061710  117737  117102  001014           MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
16038  061716  032777  020000  117114           BIT     #BIT13,@$SWR    ;;SKIP TYPEOUT IF SET
16039  061724  001004                           BNE     20$             ;;SKIP TYPEOUTS
16040  061726  004737  062046                   JSR     PC,$ERRTYP      ;;GO TO USER ERROR ROUTINE
16041  061732  104401  001115                   TYPE    ,$CRLF
16042  061736                           20$:
16043  061736  122737  000001  001140           CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
16044  061744  001007                           BNE     2$              ;;NO,SKIP APT ERROR REPORT
16045  061746  113737  001014  061760           MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
16046  061754  004737  062736                   JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
16047  061760     000                   21$:    .BYTE   0
16048  061761     000                           .BYTE   0
16049  061762  000777                   22$:    BR      22$             ;;APT ERROR LOOP
16050  061764  005777  117050           2$:     TST     @$SWR           ;;HALT ON ERROR
16051  061770  100001                           BPL     3$              ;;SKIP IF CONTINUE
16052  061772  000000                           HALT                    ;;HALT ON ERROR!
16053  061774  032777  001000  117036   3$:     BIT     #BIT09,@$SWR    ;;LOOP ON ERROR SWITCH SET?
16054  062002  001402                           BEQ     4$              ;;BR IF NO
16055  062004  013716  001010                   MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
16056  062010  005737  001112           4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
16057  062014  001402                           BEQ     5$              ;;BR IF NONE
16058  062016  013716  001112                   MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
16059  062022                           5$:
16060  062022  012737  177777  001074           MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
```

```
16061  062030  042766  000020  000002          BIC     #20,2(SP)       ;CLEAR T BIT IN CASE ERROR OCCURED
16062                                                                   ;IN T BIT TESTS
16063  062036  000002                          RTI
16064
16065  062040  005137  063240          ERRA:   COM     @#ERRFLG        ;THESE TWO INSTRUCTIONS ARE USED
16066  062044  000002                          RTI                     ;IN THE BASIC TESTS TO VERIFY THE EMT
16067
16068                                  .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
16069
16070                                  ;;**********************************************************************
16071                                  ;*THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
16072                                  ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE "ERROR TABLE" ($ERRTB),
16073                                  ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
16074
16075  062046                          $ERRTYP:
16076  062046  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16077  062052  010046                          MOV     R0,-(SP)        ;;SAVE R0
16078  062054  005000                          CLR     R0              ;;PICKUP THE ITEM INDEX
16079  062056  153700  001014                  BISB    @#$ITEMB,R0
16080  062062  001004                          BNE     1$              ;;IF ITEM NUMBER IS ZERO, JUST
16081                                                                   ;;TYPE THE PC OF THE ERROR
16082  062064  013746  001016                  MOV     $ERRPC,-(SP)    ;;SAVE $ERRPC FOR TYPEOUT
16083                                                                   ;;ERROR ADDRESS
16084  062070  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16085  062072  000426                          BR      6$              ;;GET OUT
16086  062074  005300          1$:             DEC     R0              ;;ADJUST THE INDEX SO THAT IT WILL
16087  062076  006300                          ASL     R0              ;;      WORK FOR THE ERROR TABLE
16088  062100  006300                          ASL     R0
16089  062102  006300                          ASL     R0
16090  062104  062700  001150                  ADD     #$ERRTB,R0      ;;FORM TABLE POINTER
16091  062110  012037  062120                  MOV     (R0)+,2$        ;;PICKUP "ERROR MESSAGE" POINTER
16092  062114  001404                          BEQ     3$              ;;SKIP TYPEOUT IF NO POINTER
16093  062116  104401                          TYPE                    ;;TYPE THE "ERROR MESSAGE"
16094  062120  000000          2$:             .WORD   0               ;;"ERROR MESSAGE" POINTER GOES HERE
16095  062122  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16096  062126  012037  062136          3$:     MOV     (R0)+,4$        ;;PICKUP "DATA HEADER" POINTER
16097  062132  001404                          BEQ     5$              ;;SKIP TYPEOUT IF 0
16098  062134  104401                          TYPE                    ;;TYPE THE "DATA HEADER"
16099  062136  000000          4$:             .WORD   0               ;;"DATA HEADER" POINTER GOES HERE
16100  062140  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16101  062144  011000          5$:             MOV     (R0),R0         ;;PICKUP "DATA TABLE" POINTER
16102  062146  001004                          BNE     7$              ;;GO TYPE THE DATA
16103  062150  012600          6$:             MOV     (SP)+,R0        ;;RESTORE R0
16104  062152  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16105  062156  000207                          RTS     PC              ;;RETURN
16106  062160                          7$:
16107  062160  013046                          MOV     @(R0)+,-(SP)    ;;SAVE @(R0)+ FOR TYPEOUT
16108  062162  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16109  062164  005710                          TST     (R0)            ;;IS THERE ANOTHER NUMBER?
16110  062166  001770                          BEQ     6$              ;;BR IF NO
16111  062170  104401  062176                  TYPE    ,8$             ;;TYPE TWO(2) SPACES
16112  062174  000771                          BR      7$              ;;LOOP
16113  062176  020040      000         8$:     .ASCIZ  /  /            ;;TWO(2) SPACES
16114          062202                          .EVEN
16115
16116                                  ; **********************************************************************
```

```
                                        .SBTTL PRINT ROUTINES
                                ; ******************************************************************

062202  005137  063236          PRINA:  COM     @#PRIFLG        ;THESE TWO INSTRUCTIONS ARE
062206  000002                          RTI                     ;USED BY THE BASIC TESTS TO VERIFY
                                                                ;THE TRAP INSTRUCTION

                                .SBTTL  TYPE ROUTINE

                                ;;******************************************************************
                                ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
                                ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
                                ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
                                ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
                                ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
                                ;*
                                ;*CALL:
                                ;*1) USING A TRAP INSTRUCTION
                                ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
                                ;*OR
                                ;*      TYPE
                                ;*      MESADR
                                ;*

062210  105737  001057          $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
062214  100002                  BPL     1$              ;;BR IF YES
062216  000000                  HALT                    ;;HALT HERE IF NO TERMINAL
062220  000430                  BR      3$              ;;LEAVE
062222  010046                  1$:     MOV     R0,-(SP)        ;;SAVE R0
062224  017600  000002                  MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
062230  122737  000001  001140          CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
062236  001011                  BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
062240  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
062246  001405                  BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
062250  010037  062260                  MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
062254  004737  062726                  JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
062260  000000                  61$:    .WORD   0               ;;MESSAGE ADDRESS
062262  132737  000040  001141  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
062270  001003                  BNE     60$             ;;YES,SKIP TYPE OUT
062272  112046                  2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
062274  001005                  BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
062276  005726                  TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
062300  012600                  60$:    MOV     (SP)+,R0        ;;RESTORE R0
062302  062716  000002          3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
062306  000002                  RTI                     ;;RETURN
062310  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
062314  001430                  BEQ     8$
062316  122716  000200                  CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
062322  001006                  BNE     5$
062324  005726                  TST     (SP)+           ;;POP  <CR><LF> EQUIV
062326  104401                  TYPE                    ;;TYPE A CR AND LF
062330  001115                  $CRLF
062332  105037  062466                  CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
062336  000755                  BR      2$              ;;GET NEXT CHARACTER
062340  004737  062422          5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
062344  123726  001056          6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
```

```
16173  062350  001350                        BNE     2$              ;;IF NO GO GET NEXT CHAR.
16174  062352  013746  001054                MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
16175                                                                  ;;AND THE NULL CHAR.
16176  062356  105366  000001        7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
16177  062362  002770                        BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
16178  062364  004737  062422                JSR     PC,$TYPEC       ;;GO TYPE A NULL
16179  062370  105337  062466                DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
16180  062374  000770                        BR      7$              ;;LOOP
16181
16182                                ;HORIZONTAL TAB PROCESSOR
16183
16184  062376  112716  000040        8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
16185  062402  004737  062422        9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
16186  062406  132737  000007  062466        BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
16187  062414  001372                        BNE     9$              ;;TAB STOP
16188  062416  005726                        TST     (SP)+           ;;POP SPACE OFF STACK
16189  062420  000724                        BR      2$              ;;GET NEXT CHARACTER
16190  062422  105777  116422        $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
16191  062426  100375                        BPL     $TYPEC
16192  062430  116677  000002  116414        MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
16193  062436  122766  000015  000002        CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
16194  062444  001003                        BNE     1$              ;;BRANCH IF NO
16195  062446  105037  062466                CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
16196  062452  000406                        BR      $TYPEX          ;;EXIT
16197  062454  122766  000012  000002  1$:   CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
16198  062462  001402                        BEQ     $TYPEX          ;;BRANCH IF YES
16199  062464  105227                        INCB    (PC)+           ;;COUNT THE CHARACTER
16200  062466  000000        $CHARCNT:.WORD  0                       ;;CHARACTER COUNT STORAGE
16201  062470  000207        $TYPEX: RTS     PC
16202
16203
16204                                .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
16205
16206                                ;;**********************************************************************
16207                                ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
16208                                ;*OCTAL (ASCII) NUMBER AND TYPE IT.
16209                                ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
16210                                ;*CALL:
16211                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16212                                ;*      TYPOS                   ;;CALL FOR TYPEOUT
16213                                ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
16214                                ;*      .BYTE   M               ;;M=1 OR 0
16215                                ;*                                      ;;1=TYPE LEADING ZEROS
16216                                ;*                                      ;;0=SUPPRESS LEADING ZEROS
16217                                ;*
16218                                ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
16219                                ;*$TYPOS OR $TYPOC
16220                                ;*CALL:
16221                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16222                                ;*      TYPON                   ;;CALL FOR TYPEOUT
16223                                ;*
16224                                ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
16225                                ;*CALL:
16226                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16227                                ;*      TYPOC                   ;;CALL FOR TYPEOUT
16228
```

```
 16229  062472  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
 16230  062476  116637  000001  062715          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
 16231  062504  116637  062717                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
 16232  062510  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
 16233  062514  000406                          BR      $TYPON
 16234  062516  112737  000001  062715  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
 16235  062524  112737  000006  062717          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
 16236  062532  112737  000005  062714  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
 16237  062540  010346                          MOV     R3,-(SP)        ;;SAVE R3
 16238  062542  010446                          MOV     R4,-(SP)        ;;SAVE R4
 16239  062544  010546                          MOV     R5,-(SP)        ;;SAVE R5
 16240  062546  113704  062717                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
 16241  062552  005404                          NEG     R4
 16242  062554  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
 16243  062560  110437  062716                  MOVB    R4,@#OMODE      ;;SAVE IT FOR USE
 16244  062564  113704  062715                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
 16245  062570  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
 16246  062574  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
 16247  062576  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
 16248  062600  000404                          BR      3$              ;;GO DO MSB
 16249  062602  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
 16250  062604  006105                          ROL     R5
 16251  062606  006105                          ROL     R5
 16252  062610  010503                          MOV     R5,R3
 16253  062612  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
 16254  062614  105337  062716                  DECB    $OMODE          ;;TYPE THIS DIGIT?
 16255  062620  100016                          BPL     7$              ;;BR IF NO
 16256  062622  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
 16257  062626  001002                          BNE     4$              ;;TEST FOR 0
 16258  062630  005704                          TST     R4              ;;SUPPRESS THIS 0?
 16259  062632  001403                          BEQ     5$              ;;BR IF YES
 16260  062634  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
 16261  062636  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
 16262  062642  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
 16263  062646  110337  062712                  MOVB    R3,@#8$         ;;SAVE FOR TYPING
 16264  062652  104401  062712                  TYPE    ,@#8$           ;;GO TYPE THIS DIGIT
 16265  062656  105337  062714          7$:     DECB    $OCNT           ;;COUNT BY 1
 16266  062662  003347                          BGT     2$              ;;BR IF MORE TO DO
 16267  062664  002402                          BLT     6$              ;;BR IF DONE
 16268  062666  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
 16269  062670  000744                          BR      2$              ;;GO DO THE LAST DIGIT
 16270  062672  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
 16271  062674  012604                          MOV     (SP)+,R4        ;;RESTORE R4
 16272  062676  012603                          MOV     (SP)+,R3        ;;RESTORE R3
 16273  062700  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
 16274  062706  012616                          MOV     (SP)+,(SP)
 16275  062710  000002                          RTI                     ;;RETURN
 16276  062712     000          8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
 16277  062713     000                  .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
 16278  062714     000          $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
 16279  062715     000          $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
 16280  062716  000000          $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
 16281
 16282                          .SBTTL  APT COMMUNICATIONS ROUTINE
 16283
 16284                          ;;**************************************************************************
```

```
16285  062720  112737  000001  063164  $ATY1:  MOVB    #1,$FFLG        ;;TO REPORT FATAL ERROR
16286  062726  112737  000001  063162  $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
16287  062734  000403                  BR      $ATYC
16288  062736  112737  000001  063164  $ATY4:  MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
16289  062744                          $ATYC:
16290  062744  010046                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
16291  062746  010146                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
16292  062750  105737  063162          TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
16293  062754  001450                  BEQ     5$              ;;IF NOT:  BR
16294  062756  122737  000001  001140  CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
16295  062764  001031                  BNE     3$              ;;IF NOT:  BR
16296  062766  132737  000100  001141  BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
16297  062774  001425                  BEQ     3$              ;;IF NOT:  BR
16298  062776  017600  000004          MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
16299  063002  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16300  063010  005737  001120          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
16301  063014  001375                  BNE     1$              ;;IF NOT:  WAIT
16302  063016  010037  001134          MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
16303  063022  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
16304  063024  001376                  BNE     2$
16305  063026  163700  001134          SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
16306  063032  006200                  ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
16307  063034  010037  001136          MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
16308  063040  012737  000004  001120  MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
16309  063046  000413                  BR      5$
16310  063050  017637  000004  063074  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
16311  063056  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
16312  063064  013746  177776          MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
16313  063070  004737  062210          JSR     PC,$TYPE        ;;CALL TYPE MACRO
16314  063074  000000                  4$:     .WORD   0
16315  063076                          5$:
16316  063076  105737  063164          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
16317  063102  001416                  BEQ     12$             ;;IF NOT:  BR
16318  063104  005737  001140          TST     $ENV            ;;RUNNING UNDER APT?
16319  063110  001413                  BEQ     12$             ;;IF NOT:  BR
16320  063112  005737  001120          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
16321  063116  001375                  BNE     11$             ;;IF NOT:  WAIT
16322  063120  017637  000004  001122  MOV     @4(SP),$FATAL   ;;GET ERROR #
16323  063126  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16324  063134  005237  001120          INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
16325  063140  105037  063164          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
16326  063144  105037  063163          CLRB    $LFLG           ;;CLEAR LOG FLAG
16327  063150  105037  063162          CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
16328  063154  012601                  MOV     (SP)+,R1        ;;POP STACK INTO R1
16329  063156  012600                  MOV     (SP)+,R0        ;;POP STACK INTO R0
16330  063160  000207                  RTS     PC              ;;RETURN
16331  063162     000                  $MFLG:  .BYTE   0       ;;MESSG. FLAG
16332  063163     000                  $LFLG:  .BYTE   0       ;;LOG FLAG
16333  063164     000                  $FFLG:  .BYTE   0       ;;FATAL FLAG
16334          063166                  .EVEN
16335          000200          APTSIZE=200
16336          000001          APTENV=001
16337          000100          APTSPOOL=100
16338          000040          APTCSUP=040
16339
16340                          .SBTTL  TRAP DECODER
```

```
16341
16342                                   ;;**********************************************************************
16343                                   ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
16344                                   ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
16345                                   ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
16346                                   ;*GO TO THAT ROUTINE.
16347
16348  063166  010046                  $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
16349  063170  016600  000002                  MOV     2(SP),R0        ;;GET TRAP ADDRESS
16350  063174  005740                          TST     -(R0)           ;;BACKUP BY 2
16351  063176  111000                          MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
16352  063200  006300                          ASL     R0              ;;POSITION FOR INDEXING
16353  063202  016000  063222                  MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
16354  063206  000200                          RTS     R0              ;;GO TO ROUTINE
16355
16356
16357                                   ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
16358
16359  063210  011646                  $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
16360  063212  016666  000004  000002          MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
16361  063220  000002                          RTI                     ;;RESTORE THE PSW
16362
16363                                   .SBTTL  TRAP TABLE
16364
16365                                   ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
16366                                   ;*BY THE "TRAP" INSTRUCTION.
16367
16368                                   ;       ROUTINE
16369                                   ;       -------
16370  063222  063210                  $TRPAD: .WORD   $TRAP2
16371  063224  062210                          $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
16372  063226  062516                          $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
16373  063230  062472                          $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
16374  063232  062532                          $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
16375
16376
16377
16378                                   ;FLAGS, CONSTANTS, AND VARIABLES
16379
16380  063234  000000                  BPTLOC: 0                       ;STORES 16 USER DEFINED MAINTENANCE
16381                                                                  ;BREAKPOINTS
16382  063236  000000                  PRIFLG: 0                       ;FLAG USED BY BASIC TESTS FOR TRAP TEST
16383  063240  000000                  ERRFLG: 0                       ;FLAG USED BY BASIC TESTS FOR EMT TEST
16384  063242  000000                  SELTST: 0                       ;STORES SR<8:0> FOR LOOP ON SELECTED TEST
16385  063244  000000                  SCOFLG: 0                       ;USED BY BASIC TESTS FOR IOT TEST
16386  063246  000000                  RSVFLG: 0                       ;FLAG USED BY BASIC TEST OF RSVD INSTR TRAP
16387  063250  000000                  BERFLG: 0                       ;FLAG USED BY BASIC TEST OF BUS ERROR TRAPS
16388  063252  000000                  CATERR: 0                       ;FLAGS USED BY BUS ERROR AND RSVD INSTR TRAP
16389                                                                  ;SERVICE ROUTINES
16390  063254  000000                  ONCE:   0                       ;FLAGS PROGRAM TITLE HAS BEEN PRINTED
16391                                   ;COMMON DATA STRUCTURES AND MISCELLANEOUS TABLES
16392
16393  063256  177400                  OBUF:   177400                  ;DL11 OUTPUT TEST BUFFER
16394  063260  177400                          177400
16395  063262  177400                          177400
16396  063264  177400                          177400
```

```
16397
16398  063266  000004                  IBUF:   .BLKW   4               ;DL11 INPUT TEST BUFFER
16399
16400  063276  063322                  ATA:    DWTA
16401  063300  064032                          DWTB
16402  063302  064630                          DBTA
16403  063304  064634                          DBTB
16404  063306  063312                          MBUF0
16405  063310  063316                          MBUF1
16406
16407  063312  000000                  MBUF0:  0
16408  063314  000000                          0
16409  063316  000000                  MBUF1:  0
16410  063320  000000                          0
16411  063322  000000                  DWTA:   0
16412  063324  177777                          -1
16413  063326  177400                          177400
16414  063330  000377                          377
16415  063332  125252                          125252
16416  063334  052525                  ALUADD: 052525                  ;ALSO SERVES AS NULL ENTRY FOR ALUADD
16417
16418                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ADD TEST IN THE
16419                                  ;COMBINED INSTRUCTION TESTS
16420
16421  063336  000000                          000000                  ;SRC OP1
16422  063340  000000                          000000                  ;DST OP1
16423  063342  000000                          000000                  ;ANS1
16424  063344  177777                          177777                  ;SRC OP2
16425  063346  177777                          177777                  ;DST OP2
16426  063350  177776                          177776                  ;ANS2
16427  063352  125252                          125252                  ;SRC OP3
16428  063354  052525                          052525                  ;DST OP3
16429  063356  177777                          177777                  ;ANS3
16430  063360  052525                          052525                  ;SRC OP4
16431  063362  125252                          125252                  ;DST OP4
16432  063364  177777                          177777                  ;ANS4
16433  063366  125252                          125252                  ;SRC OP5
16434  063370  125252                          125252                  ;DST OP5
16435  063372  052524                          052524                  ;ANS5
16436  063374  052525                          052525                  ;SRC OP6
16437  063376  052525                          052525                  ;DST OP6
16438  063400  125252                          125252                  ;ANS6
16439  063402  052525                          052525                  ;SRC OP7
16440  063404  125253                          125253                  ;DST OP7
16441  063406  000000                          000000                  ;ANS7
16442  063410  125253                          125253                  ;SRC OP8
16443  063412  052525                          052525                  ;DST OP8
16444  063414  000000                  ANDTAB: 000000                  ;ANS8 -- ALSO NULL ENTRY FOR ANDTAB
16445
16446                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "AND" TESTS IN THE
16447                                  ;COMBINED INSTRUCTION EXERCISER TESTS
16448
16449  063416  000000                          000000                  ;SRC OP1
16450  063420  000000                          000000                  ;DEST OP1
16451  063422  000000                          000000                  ;ANS1
16452  063424  177777                          177777                  ;SRC OP2
```

```
16453  063426  177777                    177777                  ;DST OP2
16454  063430  000000                    000000                  ;ANS2
16455  063432  000000                    000000                  ;SRC OP3
16456  063434  177777                    177777                  ;DST OP3
16457  063436  177777                    177777                  ;ANS3
16458  063440  177777                    177777                  ;SRC OP4
16459  063442  000000                    000000                  ;DST OP4
16460  063444  000000                    000000                  ;ANS4
16461  063446  125252                    125252                  ;SRC OP5
16462  063450  125252                    125252                  ;DST OP5
16463  063452  000000                    000000                  ;ANS5
16464  063454  052525                    052525                  ;SRC OP6
16465  063456  052525                    052525                  ;DST OP6
16466  063460  000000                    000000                  ;ANS6
16467  063462  125252                    125252                  ;SRC OP7
16468  063464  052525                    052525                  ;DST OP7
16469  063466  052525                    052525                  ;ANS7
16470  063470  052525                    052525                  ;SRC OP8
16471  063472  125252                    125252                  ;DST OP8
16472  063474  125252          ORTAB:    125252                  ;ANS8 -- ALSO NULL ENTRY FOR ORTAB
16473
16474                          ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "OR" TEST IN THE
16475                          ;COMBINED INSTRUCTION EXERCISER TEST
16476
16477  063476  000000                    000000                  ;SRC OP1
16478  063500  000000                    000000                  ;DEST OP1
16479  063502  000000                    000000                  ;ANS1
16480  063504  177777                    177777                  ;SRC OP2
16481  063506  177777                    177777                  ;DST OP2
16482  063510  177777                    177777                  ;ANS2
16483  063512  000000                    000000                  ;SRC OP3
16484  063514  177777                    177777                  ;DST OP3
16485  063516  177777                    177777                  ;ANS3
16486  063520  177777                    177777                  ;SRC OP4
16487  063522  000000                    000000                  ;DST OP4
16488  063524  177777                    177777                  ;ANS4
16489  063526  125252                    125252                  ;SRC OP5
16490  063530  125252                    125252                  ;DST OP5
16491  063532  125252                    125252                  ;ANS5
16492  063534  052525                    052525                  ;SRC OP6
16493  063536  052525                    052525                  ;DST OP6
16494  063540  052525                    052525                  ;ANS6
16495  063542  125252                    125252                  ;SRC OP7
16496  063544  052525                    052525                  ;DST OP7
16497  063546  177777                    177777                  ;ANS7
16498  063550  052525                    052525                  ;SRC OP8
16499  063552  125252                    125252                  ;DST OP8
16500  063554  177777          ALUSUB:   177777                  ;ANS8 -- ALSO NULL ENTRY FOR ALUSUB
16501
16502                          ;THIS TABLE OF 8 ENTRIES IS USED BYTHE ALU SUB TEST IN THE
16503                          ;COMBINED INSTRUCTION EXERCISER TESTS
16504
16505  063556  000000                    000000                  ;SRC OP1
16506  063560  000000                    000000                  ;DST OP1
16507  063562  000000                    000000                  ;ANS1
16508  063564  177777                    177777                  ;SRC OP2
```

```
16509  063566  177777                    177777                  ;DST OP2
16510  063570  000000                    000000                  ;ANS2
16511  063572  125252                    125252                  ;SRC OP3
16512  063574  052525                    052525                  ;DST OP3
16513  063576  125253                    125253                  ;ANS3
16514  063600  052525                    052525                  ;SRC OP4
16515  063602  125252                    125252                  ;DST OP4
16516  063604  052525                    052525                  ;ANS4
16517  063606  125252                    125252                  ;SRC OP5
16518  063610  125252                    125252                  ;DST OP5
16519  063612  000000                    000000                  ;ANS5
16520  063614  052525                    052525                  ;SRC OP6
16521  063616  052525                    052525                  ;DST OP6
16522  063620  000000                    000000                  ;ANS6
16523  063622  052525                    052525                  ;SRC OP7
16524  063624  125253                    125253                  ;DST OP7
16525  063626  052526                    052526                  ;ANS7
16526  063630  125253                    125253                  ;SRC OP8
16527  063632  052525                    052525                  ;DST OP8
16528  063634  125252                    125252                  ;ANS8
16529
16530  063636  005702          INSTAB:   TST     R2              ;BEGINNING OF INSTRUCTION TABLE OF INSTRUCTIONS
16531  063640  005002                    CLR     R2              ;THAT  TEST BUT SERVICE IN VARIOUS ROM LOCATIONS
16532  063642  005102                    COM     R2
16533  063644  005202                    INC     R2
16534  063646  005302                    DEC     R2
16535  063650  005502                    ADC     R2
16536  063652  005602                    SBC     R2
16537  063654  006202                    ASR     R2
16538  063656  006302                    ASL     R2
16539  063660  105002                    CLRB    R2
16540  063662  105102                    COMB    R2
16541  063664  105202                    INCB    R2
16542  063666  105302                    DECB    R2
16543  063670  105502                    ADCB    R2
16544  063672  105502                    ADCB    R2
16545  063674  105602                    SBCB    R2
16546  063676  105702                    TSTB    R2
16547  063700  106202                    ASRB    R2
16548  063702  106302                    ASLB    R2
16549  063704  151302                    BISB    (R3),R2
16550  063706  074302                    XOR     R3,R2
16551  063710  121302                    CMPB    (R3),R2
16552  063712  131302                    BITB    (R3),R2
16553  063714  141302                    BICB    (R3),R2
16554  063716  111302                    MOVB    (R3),R2
16555  063720  021302                    CMP     (R3),R2
16556  063722  031302                    BIT     (R3),R2
16557  063724  041302                    BIC     (R3),R2
16558  063726  051302                    BIS     (R3),R2
16559  063730  006702                    SXT     R2
16560  063732  005402                    NEG     R2
16561  063734  161302                    SUB     (R3),R2
16562  063736  020312                    CMP     R3,(R2)
16563  063740  030312                    BIT     R3,(R2)
16564  063742  120312                    CMPB    R3,(R2)
```

```
16565  063744  131302                  BITB    (R3),R2
16566  063746  005712                  TST     (R2)
16567  063750  105712                  TSTB    (R2)
16568  063752  021312                  CMP     (R3),(R2)
16569  063754  031312                  BIT     (R3),(R2)
16570  063756  121312                  CMPB    (R3),(R2)
16571  063760  131312                  BITB    (R3),(R2)
16572  063762  061302                  ADD     (R3),R2
16573  063764  000302                  SWAB    R2
16574  063766  160302                  SUB     R3,R2
16575  063770  060302                  ADD     R3,R2
16576  063772  010302                  MOV     R3,R2
16577  063774  011302                  MOV     (R3),R2
16578  063776  110302                  MOVB    R3,R2
16579  064000  006102                  ROL     R2
16580  064002  106102                  ROLB    R2
16581  064004  105402                  NEGB    R2
16582  064006  102400                  BVS     .+2
16583  064010  102000                  BVC     .+2
16584  064012  000005                  RESET
16585  064014  020302                  CMP     R3,R2
16586  064016  030302                  BIT     R3,R2
16587  064020  040302                  BIC     R3,R2
16588  064022  120302                  CMPB    R3,R2
16589  064024  130302                  BITB    R3,R2
16590  064026  140302                  BICB    R3,R2
16591  064030  150302                  BISB    R3,R2
16592
16593  064032  000000          DWTB:   0                       ;ALSO SERVES AS INSTAB TABLE TERMINATOR
16594  064034  000001                  1
16595  064036  000400                  400
16596  064040  177401                  177401
16597  064042  052526                  52526
16598  064044  125253                  125253
16599                         ;*        MED TEST TABLES
16600
16601  064046  000000          TLOC1:  .WORD   0
16602  064050  000000          PSWHOL: .WORD   0
16603  064052  000000          TABBEG: .WORD   0
16604  064054  000000          TABEND: .WORD   0
16605  064056  000040          STGBLK: .BLKW   40
16606  064156  000000          VADR:   .WORD   0
16607  064160  000000          PA1716: .WORD   0
16608  064162  000000          PA1500: .WORD   0
16609  064164  000000          TLOC2:  .WORD   0
16610
16611                         ;*
16612                         ;*        TABLE II
16613                         ;*
16614                         ;*        FOLLOWING IS A TABLE OF INTERNAL REGISTER OPERATION CODES
16615                         ;*        USED FOR TESTING THE MED INSTRUCTION.  LABELS CORRESPOND
16616                         ;*        TO REGISTER NAMES, THE HIGH BYTE IS THE READ OPERATION
16617                         ;*        CODE, THE LOW BYTE THE WRITE CODE.
16618                         ;*                NOTE:  WHEN ADDING OR DELETING
16619                         ;*                ENTRIES IN THIS TABLE, CHECK DUAL
16620                         ;*                ADDRESING TEST TO SEE THAT THE "SCRATCH
```

```
                                        ;*              PAD LIMITS" ARE MAINTAINED.
                                        ;*
                                        ;*
064166                                  TBL2:
                                        ;
064166                                  ASP1:                                   ;A SCRATCH PAD - LO
064166      201     001                 R1A:    .BYTE   201,001                 ;LOBYTE, HIBYTE=WRITE CODE, READ CODE
064170      202     002                 R2A:    .BYTE   202,002
064172      203     003                 R3A:    .BYTE   203,003
064174      204     004                 R4A:    .BYTE   204,004
064176      205     005                 R5A:    .BYTE   205,005
064200      206     006                 R6A:    .BYTE   206,006
064202      210     010                 FAC3.0: .BYTE   210,010
064204      211     011                 FAC3.1: .BYTE   211,011
064206      212     012                 FAC3.2: .BYTE   212,012
064210      213     013                 FAC3.3: .BYTE   213,013
064212      214     014                 FAC3.4: .BYTE   214,014
064214      215     015                 FAC3.5: .BYTE   215,015
064216      216     016                 UR6A:   .BYTE   216,016
064220      217     017                 FDST3:  .BYTE   217,017
064222      220     020                 WCSA.0: .BYTE   220,020                 ;A SCRATCH PAD-HI
064224      221     021                 WCSA.1: .BYTE   221,021
064226      222     022                 GNUHAM: .BYTE   222,022
064230      223     023                 CNSTSW: .BYTE   223,023
064232      226     026                 CNSSW:  .BYTE   226,026
064234      227     027                 CNSCOR: .BYTE   227,027
064236      230     030                 FAC1.0: .BYTE   230,030
064240      231     031                 FAC1.1: .BYTE   231,031
064242      232     032                 FAC1.2: .BYTE   232,032
064244      233     033                 FAC1.3: .BYTE   233,033
064246      234     034                 FAC1.4: .BYTE   234,034
064250      235     035                 FAC1.5: .BYTE   235,035
064252      236     036                 FPSHI:  .BYTE   236,036
064254      237     037         ASP2:   FDST1:  .BYTE   237,037
                                        ;
064256                                  BSP1:
064256      241     041                 R1B:    .BYTE   241,041                 ;B SCRATCH PAD - LO
064260      242     042                 R2B:    .BYTE   242,042
064262      243     043                 R3B:    .BYTE   243,043
064264      244     044                 R4B:    .BYTE   244,044
064266      245     045                 R5B:    .BYTE   245,045
064270      246     046                 R6B:    .BYTE   246,046
064272      250     050                 FAC2.0: .BYTE   250,050
064274      251     051                 FAC2.1: .BYTE   251,051
064276      252     052                 FAC2.2: .BYTE   252,052
064300      253     053                 FAC2.3: .BYTE   253,053
064302      254     054                 FAC2.4: .BYTE   254,054
064304      255     055                 FAC2.5: .BYTE   255,055
064306      256     056                 UR6B:   .BYTE   256,056
064310      257     057                 FDST2:  .BYTE   257,057
064312      260     060                 WCSB.0: .BYTE   260,060                 ;B SCRATCH PAD - HI
064314      261     061                 WCSB.1: .BYTE   261,061
064316      262     062                 WCSADR: .BYTE   262,062
064320      263     063                 RZERO:  .BYTE   263,063
064322      266     066                 RVECT:  .BYTE   266,066
064324      270     070                 FAC0.0: .BYTE   270,070
```

```
16677  064326  272  072          FACO.1: .BYTE   272,072
16678  064330  273  073          FACO.2: .BYTE   273,073
16679  064332  274  074          FACO.4: .BYTE   274,074
16680  064334  275  075          FACO.5: .BYTE   275,075
16681  064336  276  076          FEA:    .BYTE   276,076
16682  064340  277  077          BSP2:   FOSTO:  .BYTE   277,077
16683
16684  064342                      CSP1:
16685  064342  300  100          LJAM:   .BYTE   300,100          ;C SCRATCH PAD
16686  064344  301  101          LSERV:  .BYTE   301,101
16687  064346  302  102          LPBA:   .BYTE   302,102
16688  064350  303  103          LCUA:   .BYTE   303,103
16689  064352  304  104          LFGIN:  .BYTE   304,104
16690  064354  305  105          LWHAM:  .BYTE   305,105
16691  064356  307  107          LTAG:   .BYTE   307,107
16692  064360  310  110          CNSC0:  .BYTE   310,110
16693  064362  311  111          CNSC1:  .BYTE   311,111
16694  064364  312  112          CNSC2:  .BYTE   312,112
16695  064366  313  113          CST200: .BYTE   313,113
16696  064370  316  116          CSP2:   CNST0:  .BYTE   316,116
16697  064372  224  024          RT1A:   .BYTE   224,024
16698  064374  225  025          RT2A:   .BYTE   225,025
16699  064376  264  064          RT1B:   .BYTE   264,064
16700  064400  265  065          RT2B:   .BYTE   265,065
16701  064402  000000                   .WORD   0
16702
16703                           ;*
16704                           ;*      TABLE III
16705                           ;*
16706                           ;*      THE FOLLOWING IS A LIST OF "NOP" OPERATION CODES
16707                           ;*      THAT WILL BE USED WITH A MED IN MED TEST 3 TO
16708                           ;*      ENSURE THAT A MED WITH THESE CODES WILL NOT HANG.
16709
16710  064404                      TBL3:
16711  064404  120  137          NOPS:   .BYTE   120,137          ;GROUP A
16712  064406  145  145                  .BYTE   145,145          ;GROUP B
16713  064410  150  151                  .BYTE   150,151          ;GROUP C
16714  064412  156  177                  .BYTE   156,177          ;GROUP D
16715  064414  320  343                  .BYTE   320,343          ;GROUP E
16716  064416  353  357                  .BYTE   353,357          ;FROUP G
16717  064420  000000                     .WORD   0               ;A 0 TERMINATES TABLE
16718
16719                           ;*
16720                           ;*      TABLE IV
16721                           ;*
16722                           ;*      THE LIST BELOW CONTAINS THOSE OPERATION CODES
16723                           ;*      CORRESPONDING TO THE INTERNAL REGISTERS WHICH MUST
16724                           ;*      BE TESTED SEPERATELY BECAUSE THEY ARE READ-ONLY,
16725                           ;*      WRITE-ONLY, OR USED IN MACRO CODE EXECUTION, ETC. . .
16726
16727  064422                      TBL4:
16728  064422  200  000          R0A:    .BYTE   200,000          ;LOBYTE,HYBYTE - WRITE CODE, READ CODE
16729  064424  207  007          R7A:    .BYTE   207,007          ;0 REPLACES ANY NON EXSISTENT CODES
16730  064426  240  040          R0B:    .BYTE   240,040          ;EXCEPT IN THE CASE OF R0A
16731  064430  247  047          R7B:    .BYTE   247,047
16732  064432  314  114          CNST2:  .BYTE   314,114
```

```
16733  064434  317  117          CNST1:  .BYTE  317,117
16734                            ;*      TABLE V
16735                            ;*
16736  064436                    TBL5:
16737
16738  064436  306               LCDTA:  .BYTE  306          ;THIS TABLE CONTAINS THE OPERATION
16739  064437  106                       .BYTE  106          ;CODES OF THOSE INTERNAL REGISTERS
16740  064440  315               MD:     .BYTE  315          ;WHICH MUST BE TESTED USING THE
16741  064441  115                       .BYTE  115          ;MICROBREAK REGISTER.  THEIR
16742  064442  267               CNSCTL: .BYTE  267          ;ASSOCIATED MICRO-ADDRESSES ARE IN
16743  064443  067                       .BYTE  067          ;THE NEXT TABLE
16744  064444  140               JAM:    .BYTE  140
16745  064445  141               SERV:   .BYTE  141
16746  064446  142               PBA:    .BYTE  142
16747  064447  143               CUA:    .BYTE  143
16748  064450  344               FLAG:   .BYTE  344
16749  064451  144                       .BYTE  144
16750  064452  345               DREG:   .BYTE  345
16751  064453  146               REV:    .BYTE  146
16752  064454  346               SREG:   .BYTE  346
16753  064455  147               COUNT:  .BYTE  147
16754  064456  347               NUA:    .BYTE  347
16755  064457  351               RES:    .BYTE  351
16756  064460  152               DCS0:   .BYTE  152
16757  064461  352                       .BYTE  352          ;INIT REG
16758  064462  153               DCS1:   .BYTE  153
16759  064463  000                       .BYTE  0            ;TABLE TERMINATOR
16760                                    .EVEN
16761
16762                            ;*      TABLE VI
16763                            ;*
16764  064464                    TBL6:
16765
16766  064464  003330            ULCDTA: .WORD  3330         ;THIS TABLE CONTAINS THE MICRO-ADDRESSES
16767  064466  003150                    .WORD  3150         ;WHICH ARE LOADED INTO THE MICROBREAK
16768  064470  003375            UMD:    .WORD  3375         ;REG. TO TEST THE OPERATION CODES
16769  064472  003271                    .WORD  3271         ;CONTAINED IN THE PRECEEDING TABLE.
16770  064474  003240            UCNSCTL:.WORD  3240
16771  064476  003224                    .WORD  3224
16772  064500  003160            UJAM:   .WORD  3160
16773  064502  003161            USERV:  .WORD  3161
16774  064504  003170            UPBA:   .WORD  3170
16775  064506  003171            UCUA:   .WORD  3171
16776  064510  003344            UFLAG:  .WORD  3344
16777  064512  003320                    .WORD  3320
16778  064514  003345            UDREG:  .WORD  3345
16779  064516  003340            UREV:   .WORD  3340
16780  064520  003350            USREG:  .WORD  3350
16781  064522  003341            UCOUNT: .WORD  3341
16782  064524  003351            UNUA:   .WORD  3351
16783  064526  003355            URES:   .WORD  3355
16784  064530  003720            UDCS0:  .WORD  3720
16785  064532  003724            UINIT:  .WORD  3724
16786  064534  003721            UDCS1:  .WORD  3721
16787
16788                            ;*      TABLE VII
```

```
16789                                   ;*
16790                                   ;*      THIS TABLE HOLDS THE OPERATION CODES AND THE CONSTANT
16791                                   ;*      VALUE EXPECTED FOR CERTAIN INTERNAL REGISTERS.
16792  064536                           TBL7:
16793
16794  064536  000100  077600          CLJAM:  .WORD   100,77600
16795  064542  000101  000010          CLSERV: .WORD   101,10
16796  064546  000102  020000          CLPBA:  .WORD   102,20000
16797  064552  000103  000004          CLCUA:  .WORD   103,4
16798  064556  000104  050000          CLFGIN: .WORD   104,50000
16799  064562  000105  054000          CLWHAM: .WORD   105,54000
16800  064566  000107  024000          CLTAG:  .WORD   107,24000
16801  064572  000110  177400          CCNSC0: .WORD   110,177400
16802  064576  000111  177600          CCNSC1: .WORD   111,177600
16803  064602  000112  100000          CCNSC2: .WORD   112,100000
16804  064606  000113  000200          CCST200:.WORD   113,200
16805  064612  000114  000002          CCNST2: .WORD   114,2
16806  064616  000116  000000          CCNST0: .WORD   116,0
16807  064622  000117  000001          CCNST1: .WORD   117,1
16808  064626  000000                          .WORD   0
16809
16810                                   .EVEN
16811  064630                           DBTA:
16812  064630     000     377     252  .BYTE   000,377,252,125
16813  064633     125
16814  064634                           DBTB:
16815  064634     000     001     120  .BYTE   000,001,120,253
16816  064637     253
16817
16818                                   ;MESSAGE TABLES
16819
16820  064640                           EM1:
16821  064640                           EM2:
16822  064640  027523  020102  051504  EM4:    .ASCII  'S/B DST '
16823  064646  020124
16824  064650  040527  020123  051504  EM7:    .ASCII  'WAS DST '
16825  064656  020124
16826  064660  042040  051505  004524  EM6:    .ASCII  ' DEST'<HT>
16827  064666  024040  051111  004451  EM5:    .ASCIZ  ' (IR)'<HT>' TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16828  064674  052040  051505  004524
16829  064702  024040  041520  004451
16830  064710  024040  050123  004451
16831  064716  050050  053523  000051
16832  064724  027523  020102  042522  EM10:   .ASCIZ  'S/B RES WAS RES DST OP  SRC OP  TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16833  064732  020123  040527  020123
16834  064740  042522  020123  051504
16835  064746  020124  050117  020040
16836  064754  051123  020103  050117
16837  064762  020040  042524  052123
16838  064770  020011  050050  024503
16839  064776  020011  051450  024520
16840  065004  024011  051520  024527
16841  065012     000
16842  065013     123  041057  051440  EM3:    .ASCIZ  'S/B SP'<HT>'WAS SP'<HT>' (IR)'<HT>' TEST'<HT>' (PC)'<HT>'(PSW)'
16843  065020  004520  040527  020123
16844  065026  050123  020011  044450
```

```
16845  065034  024522  020011  042524
16846  065042  052123  020011  020050
16847  065050  024503  024011  051520
16848  065056  024527     000
16849  065061     011  020011  051511  DH2:    .ASCIZ  <HT><HT>' IS R3'
16850  065066  051040  000063
16851  065072  004411  044440  020123  DH4:    .ASCIZ  <HT><HT>' IS R5'
16852  065100  032522     000
16853  065103     015  042412  042116  EOP1:   .ASCIZ  <15><12>'END PASS # '
16854  065110  050040  051501  020123
16855  065116  020043     000
16856  065121     011  051105  047522  EOP2:   .ASCIZ <HT>'ERROR COUNT = '
16857  065126  020122  047503  047125
16858  065134  020124  020075     000
16859  065141     015  046412  026504  IDENT1: .ASCIZ <15><12>'MD-11-DQKDA-B   KD11-K BASIC LOGIC TESTS'<15><12>
16860  065146  030461  042055  045521
16861  065154  040504  041055  020040
16862  065162  045440  030504  026461
16863  065170  020113  040502  044523
16864  065176  020103  047514  044507
16865  065204  020103  042524  052123
16866  065212  006523  000012
16867  065216  005015  051124  050101  BEMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 4 PC = '
16868  065224  042520  020104  047524
16869  065232  032040  050040  020103
16870  065240  020075     000
16871  065243     015  052012  040522  RSMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 10 PC = '
16872  065250  050120  042105  052040
16873  065256  020117  030061  050040
16874  065264  020103  020075     000
16875  065271     124  051505  051524  EM11:   .ASCIZ  'TESTS SKIPPED'
16876  065276  051440  044123  050120
16877  065304  042105     000
16878  065307     040  050040  004503  DH11:   .ASCIZ  "  PC"<HT>"EXPCTD"<HT>"ACTUAL"<HT>"(TEST #'S)"
16879  065314  054105  041520  042124
16880  065322  040411  052103  040525
16881  065330  004514  052050  051505
16882  065336  020124  023443  024523
16883  065344     000
16884  065345     115  042105  042040  EM12:   .ASCIZ  /MED DID NOT ABORT IN USER MODE/
16885  065352  042111  047040  052117
16886  065360  040440  047502  052122
16887  065366  044440  020116  051525
16888  065374  051105  046440  042117
16889  065402  000105
16890  065404  042515  020104  054105  EM13:   .ASCIZ  /MED EXECUTED IN USER MODE/
16891  065412  041505  052125  042105
16892  065420  044440  020116  051525
16893  065426  051105  046440  042117
16894  065434  000105
16895  065436  042515  020104  044103  EM14:   .ASCIZ  /MED CHANGED PSW/
16896  065444  047101  042507  020104
16897  065452  051520  000127
16898  065456  044515  051103  041117  EM15:   .ASCIZ  /MICROBREAK TRAP-TO-4 DID NOT OCCUR/
16899  065464  042522  045501  052040
16900  065472  040522  026520  047524
```

```
16901  065500  032055  042040  042111
16902  065536  047040  052117  047440
16903  065514  041503  051125     000
16904  065521     114  043517  052503  EM17:   .ASCIZ  /LOGCUA LOGGED WRONG/
16905  065536  020101  047514  043507
16906  065554  042105  053440  047522
16907  065562  043516     000
16908  065565     103  050123  041440  EM21:   .ASCIZ  /CSP CONSTANT WRONG/
16909  065572  047117  052123  047101
16910  065600  020124  051127  047117
16911  065606  000107
16912  065610  040502  020104  040504  EM22:   .ASCIZ  /BAD DATA READ BY A MED/
16913  065624  040524  051040  040505
16914  065630  020104  054502  040440
16915  065612  046440  042105     000
16916  065617     116  020117  042117  EM23:   .ASCIZ  /NO ODD PC TRAP/
16917  065624  020104  041520  052040
16918  065632  040522  000120
16919  065636  042117  020104  042101  EM24:   .ASCIZ  /ODD ADR. BIT NOT SET IN CPU ERR REG OR LOG JAM/
16920  065644  027122  041040  052111
16921  065652  047040  052117  051440
16922  065660  052105  044440  020116
16923  065666  020103  020125  051105
16924  065674  020122  042522  020107
16925  065702  051117  046040  043517
16926  065710  045040  046501     000
16927  065715     120  054510  020123  EM26:   .ASCIZ  /PHYS BA LOGGED WRONG/
16928  065722  040502  046040  043517
16929  065730  042507  020104  051127
16930  065736  047117  000107
16931  065742  040503  044103  020105  EM27:   .ASCIZ  /CACHE PARITY ERROR LOGGED IN BAKUP MODE/
16932  065750  040520  044522  054524
16933  065756  042440  051122  051117
16934  065764  046040  043517  042507
16935  065772  020104  047111  041040
16936  066000  045501  050125  046440
16937  066006  042117  000105
16938  066012  040503  044103  020105  EM30:   .ASCIZ  /CACHE PARITY TRAPPED WHEN DISABLED/
16939  066020  040520  044522  054524
16940  066026  052040  040522  050120
16941  066034  042105  053440  042510
16942  066042  020116  044504  040523
16943  066050  046102  042105     000
16944  066055     111  051516  051124  EM41:   .ASCIZ  /INSTR. NOT ABORTED IN CACHE ABORT MODE/
16945  066062  020056  047516  020124
16946  066070  041101  051117  042524
16947  066076  020104  047111  041440
16948  066104  041501  042510  040440
16949  066112  047502  052122  046440
16950  066120  042117  000105
16951  066124  042515  047515  054522  EM32:   .ASCIZ  /MEMORY ERR REG INCORRECT/
16952  066132  042440  051122  051040
16953  066140  043505  044440  041516
16954  066146  051117  042522  052103
16955  066154     000
16956  066155     124  046511  047505  EM33:   .ASCIZ  /TIMEOUT BIT NOT SET IN CPU ERR REG OR LOG JAM/
```

```
16957  066162  052125  041040  052111
16958  066170  047040  052117  051440
16959  066176  052105  044440  020116
16960  066204  050103  020125  051105
16961  066212  020122  042522  020107
16962  066220  051117  046440  043517
16963  066226  045040  046501     000
16964  066233     116  020117  046111  EM34:   .ASCIZ  /NO ILLEGAL INTERNAL ADR TRAP/
16965  066240  042514  040507  020114
16966  066246  047111  042524  047122
16967  066254  046101  040440  051104
16970  066276  020114  042101  020122
16971  066304  051105  020122  044502
16972  066312  020124  047516  020124
16973  066320  042523  020124  047111
16974  066336  041440  052520  042440
16975  066344  051122  051040  043505
16976  066342  047440  020122  047514
16977  066360  020107  040512  000115
16978  066366  040514  052123  044440  EM36:   .ASCIZ  "LAST INTR/TRAP VECTOR NOT LOGGED IN FLAG REG"
16979  066414  052116  027522  051124
16980  066372  050101  053040  041505
16981  066400  047524  020122  047516
16982  066406  020124  047514  043507
16983  066414  042105  044440  020116
16984  066422  046106  043501  051040
16985  066430  043505     000
16986  066433     114  043517  043040  EM37:   .ASCIZ  /LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR/
16987  066440  051111  052123  046440
16988  066446  042117  020105  044504
16989  066454  020104  047516  020124
16990  066462  047111  044510  044502
16991  066470  020124  051105  047522
16992  066476  020122  047514  020107
16993  066504  043101  042524  020122
16994  066512  044506  051522  020124
16995  066520  051105  047522  000122
16996  066526  051105  047522  020122  EM40:   .ASCIZ  /ERROR LOG WAS NOT REENABLED, ODD ADR BIT CLR IN CPUERR/
16997  066534  047514  020107  040527
16998  066542  020123  047516  020124
16999  066550  042522  047105  041101
17000  066556  042514  026104  047440
17001  066564  042104  040440  051104
17002  066572  041040  052111  041440
17003  066600  051114  044440  020116
17004  066606  050103  042525  051122
17005  066614     000
17006  066615     116  020117  040503  EM31:   .ASCIZ  /NO CACHE PARITY TRAP/
17007  066622  044103  020105  040520
17008  066630  044522  054524  052040
17009  066636  040522  000120
17010  066642  047514  023040  044040  EM42:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE/
17011  066650  020111  054502  042524
17012  066656  023040  052040  043501
```

```
17013  066664  050040  051101  052111
17014  066672  020131  044502  051524
17015  066700  047040  052117  051440
17016  066706  052105  044440  020116
17017  066714  047514  020107  042523
17018  066722  053122  041511  000105
17019  066730  047514  023040  044040  EM43:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN MEM ERR REG/
17020  066736  020111  054502  042524
17021  066744  023040  052040  043501
17022  066752  050040  051101  052111
17023  066760  020131  044502  051524
17024  066766  047040  052117  051440
17025  066774  052105  044440  020116
17026  067002  042515  020115  051105
17027  067010  020122  042522  000107
17030  067016  040503  044103  020105  EM45:   .ASCIZ  /CACHE TAG LOGGED WRONG/
17029  067024  040524  020107  047514
17030  067032  043507  042105  053440
17031  067040  047522  043516     000
17032  067045     103  041501  042510  EM16:   .ASCIZ  /CACHE DATA LOGGED WRONG/
17033  067052  042040  052101  020101
17034  067060  047514  043507  042105
17035  067066  053440  047522  043516
17036  067074     000
17037  067075     105  051511  051440  EMEIS1: .ASCIZ  'EIS SET COND CODES WRONG'
17038  067102  052105  041440  047117
17039  067110  020104  047503  042504
17040  067116  020123  051127  047117
17041  067124  000107
17042  067126  044505  020123  040507  EMEIS2: .ASCIZ  'EIS GAVE WRONG RESULT'
17043  067134  042522  053440  047522
17044  067142  043516  051040  051505
17045  067150  046125  000124
17046  067154  052501  047524  044455  EM46:   .ASCIZ  'AUTO-INCREMENT (DECREMT) DID NOT OCCUR IN EIS'
17047  067162  041516  042522  042515
17048  067170  052116  024040  042504
17049  067176  051103  046505  024524
17050  067204  042040  042111  047040
17051  067212  052117  047440  041503
17052  067220  051125  044440  020116
17053  067226  044505  000123
17054  067232  050040  053523  051011  DHEIS1: .ASCII  ' PSW'<HT>'REG-WAS-REG+1'<HT>'REG-S/B-REG+1'<HT>
17055  067240  043505  053455  051501
17056  067246  051055  043505  030453
17057  067254  051011  043505  051455
17058  067262  041057  051055  043505
17059  067270  030453     011
17060  067273     040  050040  004503  DH46:   .ASCIZ  '  PC'<HT>' (IR)'<HT>' TEST'
17061  067300  024040  051111  004451
17062  067306  052040  051505  000124
17063  067314  020040  041520  046411  DH15:   .ASCIZ  /  PC/<HT>/MEDCODE MICROBK REG./
17064  067322  042105  047503  042504
17065  067330  046440  041511  047522
17066  067336  045502  051040  043505
17067  067344  000056
17068  067346  020040  041520  046411  DH17:   .ASCIZ  /  PC/<HT>/MEDCODE EXPECTD RECEIVD/
```

```
17069  067354  042105  047503  042504
17070  067360  042440  050130  041505
17071  067366  042124  051040  041505
17072  067376  044515  042126     000
17073  067403     040  050040  000103  DH23:   .ASCIZ  /  PC/
17074  067410  020040  041520  041411  DH24:   .ASCIZ  /  PC/<HT>/CPUERR/<HT>/LOGJAM/
17075  067416  052520  051105  004522
17076  067424  047514  045107  046501
17077  067432     000
17078  067433     040  050040  004503  DH25:   .ASCIZ  /  PC/<HT>/FLGREG/
17079  067440  046106  051107  043505
17080  067446     000
17081  067447     040  050040  004503  DH26:   .ASCIZ  '  PC'<HT>'<17:16>-S/B PA-<15:0>  <17:16>-WAS PA-<15:0>'
17082  067454  030474  020057  033061
17083  067462  026476  027523  020102
17084  067470  040520  036055  032461
17085  067476  030072  020076  036040
17086  067504  033461  030472  037066
17087  067512  053455  051501  050040
17088  067520  026501  030474  035065
17089  067526  037060     000
17090  067531     040  050040  004503  DH27:   .ASCIZ  /  PC/<HT>/LOGPBA/<HT>/LOGDATA/<HT>/LOGTAG/
17091  067536  047514  050107  040502
17092  067544  046011  043517  040504
17093  067552  040524  046011  043517
17094  067560  040524  000107
17095  067564  020040  041520  046411  DH32:   .ASCIZ  /  PC/<HT>/MEMERR/
17096  067572  046505  051105  000122
17097  067600  020040  041520  046011  DH44:   .ASCIZ  /  PC/<HT>/LOGSERVCE/
17098  067606  043517  042523  053122
17099  067614  042503     000
17100  067617     040  050040  004503  DH44:   .ASCIZ  /  PC/<HT>/EXPCT/<HT>/RECVD/
17101  067624  054105  041520  004524
17102  067632  042522  053103  000104
17103                                  .EVEN
17104  067640  001016  001076  001100  DT15:   .WORD   SERRPC,STMP0,STMP1,0
17105  067646  000000
17106  067650  001016  001100  001102  DT21:   .WORD   SERRPC,STMP1,STMP2,SREG0,0
17107  067656  001062  000000
17108  067662  001016  001100  001102  DT22:   .WORD   SERRPC,STMP1,STMP2,STMP3,0
17109  067670  001104  000000
17110  067674  001016  000000          DT23:   .WORD   SERRPC,0
17111  067700  001016  001064  001062  DT24:   .WORD   SERRPC,SREG1,SREG0,0
17112  067706  000000
17113  067710  001016  001062  000000  DT25:   .WORD   SERRPC,SREG0,0
17114  067716  001016  001064  001066  DT26:   .WORD   SERRPC,SREG1,SREG2,SREG0,SREG3,0
17115  067724  001062  001070  000000
17116  067732  001016  001070  001064  DT27:   .WORD   SERRPC,SREG3,SREG1,SREG2,0
17117  067740  001066  000000
17118  067744  001060  001066  001070  DTEIS1: .WORD   SREGAD,SREG2,SREG3,SREG1,SREG4
17119  067752  001064  001072
17120  067756  001016  001076  001062  DT46:   .WORD   SERRPC,STMP0,SREG0,0
17121  067764  000000
17122
17123  067766     000     000          DF15:   .BYTE   0,0
17124  067770     000     000     000  DF17:   .BYTE   0,0,0
```

```
17125            067774                    .EVEN
17126  067774                              DT1:
17127  067774                              DT2:
17128  067774                              DT4:
17129  067774   001072                     DT10:   .WORD   $REG4
17130  067776   001070                     DT7:    .WORD   $REG3
17131  070000   001066                     DT6:    .WORD   $REG2
17132  070002   001064   001062   001016   DT5:    .WORD   $REG1,$REG0,$ERRPC,$REG5,$REGAD,0
17133  070010   001074   001060   000000
17134  070016   001072   001070   001064   DT3:    .WORD   $REG4,$REG3,$REG1,$REG0,$ERRPC,$REGAD,0
17135  070024   001062   001016   001060
17136  070032   000000
17137  070034   001016   001124   001062   DT11:   .WORD   $ERRPC,$TESTN,$REG0,0
17138  070042   000000
17139            000001                    .END
```

ABASE =	000000	937											
ACOM1 =	000010	937											
ACOM2 =	000000	937											
ACPUOP=	000000	937	952										
ADDM0 =	000000	937											
ADDM1 =	000000	937											
ADDM10=	000000	937											
ADDM11=	000000	937											
ADDM12=	000000	937											
ADDM13=	000000	937											
ADDM14=	000000	937											
ADDM15=	000000	937											
ADDM2 =	000000	937											
ADDM3 =	000000	937											
ADDM4 =	000000	937											
ADDM5 =	000000	937											
ADDM6 =	000000	937											
ADDM7 =	000000	937											
ADDM8 =	000000	937											
ADDM9 =	000000	937											
ADEVCT=	000000	937	943										
ADEVM =	000000	937											
AEMV =	000000	937	948										
AEMVM =	000000	937	949										
AFATAL=	000000	937	940										
ALUADD	063334	14011	14016	16416#									
ALUSUB	063554	14071	14076	16500#									
AMADR1=	000000	937											
AMADR2=	000000	937											
AMADR3=	000000	937											
AMADR4=	000000	937											
AMAMS1=	000000	937											
AMAMS2=	000000	937											
AMAMS3=	000000	937											
AMAMS4=	000000	937											
AMSCAD=	000000	937	945										
AMSCLK=	000000	937	946										
AMSCTY=	000000	937	939										
AMTYP1=	000000	937											
AMTYP2=	000000	937											
AMTYP3=	000000	937											
AMTYP4=	000000	937											
AMOTAB	063414	14131	14136	16444#									
APASS =	000000	937	942										
APRIOR=	000000	937											
APTCSU=	000040	16154	16338#										
APTENV=	000001	16043	16147	16294	16336#								
APTSIZ=	000200	16335#											
APTSPO=	000100	16149	16296	16337#									
ASP1	064166	16626#											
ASP2	064254	16654#											
ASWREG=	000000	937	950										
ATA	063276	2645	2695	2707	2723	2755	2768	2816	2837	2856	2868	2878	2890	2901
		7636	7647	7660	7671	7684	7696	7709	7720	7733	7745	7758	7777	7798
		7819	7840	7869	7898	7899	7913	7927	7928	7942	7956	7985	8014	8015
		8029	8043	8044	8058	8072	8094	8116	8138	8776	8788	8828	8840	8875

```
                        9100    9114    9137    9151    9252    9266    9289    9303    9390    9422    9494    9536    9771
                        9776    9825    9830    9874    9939   11088   11131   11175   14413   14433   14485   14505   14552
                       16400#
ATESTN= 000000           937     941
AUNIT = 000000           937     944
AUSWR = 000000           937     951
AVECT1= 000000           937
AVECT2= 000000           937
A002    001636          1289#    1293
A015    002106          1455     1460#
A017    002200          1498     1505#
A020    002246          1527     1532#
A021    002276          1549     1554#
A023    002364          1590     1595#
A032    002570          1695     1700#
A033    002630          1717     1722#
A034    002670          1738     1743#
A041    003034          1819     1824#
A046    003234          1917     1922#
BASIC   003246          1923     1931#
BEMSG   065216         15914    16867#
BERFLG  063250          4472#    4478#  15905#   16387#
BERR    061220          4510    12319   12323    13187   13225   13264   13281   13333   13393   13431   13462   13494   14921
                       15148    15174   15340    15370   15418   15546   15593   15624   15640   15659   15674   15908#
BETST   061212          4469    15905#
BIT0  = 000001           819#    1226    2301    15165   15422   15459   15540   15577   15614   15641
BIT00 = 000001           809#     819
BIT01 = 000002           808#     818
BIT02 = 000004           807#     817
BIT03 = 000010           806#     816
BIT04 = 000020           805#     815
BIT05 = 000040           804#     814
BIT06 = 000100           803#     813
BIT07 = 000200           802#     812
BIT08 = 000400           801#     811
BIT09 = 001000           800#     810    15968    16053
BIT1  = 000002           818#    2962    15490
BIT10 = 002000           799#   10030
BIT11 = 004000           798#   10900    15046    15975
BIT12 = 010000           797#   11712
BIT13 = 020000           796#   12766    15351    16038
BIT14 = 040000           795#   13906    15950
BIT15 = 100000           794#   14857    15096    15123   15165   15184   15459   15500   15577   15607   15614   15852
BIT2  = 000004           817#    3694    15184    15382   15607
BIT3  = 000010           816#    4203
BIT4  = 000020           815#    4928    15344    15597
BIT5  = 000040           814#    1230     5801    15428
BIT6  = 000100           813#    1225     1228     6653   15178   15374   15625
BIT7  = 000200           812#    1227     1229     7509   15351   15527
BIT8  = 000400           811#    8324
BIT9  = 001000           810#    9245    15091    15142   15351
BKFLAG  061112         15088#   15100    15120#   15130   15795#  15798#
BKROUT  061104         15086    15795#
BPTLOC  063234          2301    2962     3694     4203    4928    5801    6653    7509    8324    9245   10030   10900   11712
                       12766   13906    14857    16380#
BPTVEC= 000014           826#
```

[illegible]	[illegible]	16656#												
[illegible]	[illegible]	16682#												
BT001	001630	1279#												
BT002	001634	1279	1287#											
BT003	001646	1289	1301#	1308										
BT004	001662	1304	1314#	1323										
BT005	001700	1319	1329#	1338										
BT006	001716	1334	1344#	1351										
BT007	001730	1348	1357#	1366										
BT010	001744	1362	1372#	1381										
BT011	001762	1378	1387#	1397										
BT012	002002	1394	1403#	1412										
BT013	002022	1409	1418#	1429										
BT014	002046	1426	1435#	1444										
BT015	002070	1441	1450#	1458	1465									
BT016	002116	1462	1471#	1483										
BT017	002144	1479	1489#	1503	1512									
BT020	002220	1507	1518#	1530	1536									
BT021	002256	1533	1542#	1552	1559									
BT022	002310	1556	1565#	1575										
BT023	002334	1572	1581#	1593	1599									
BT024	002374	1596	1605#	1614										
BT025	002420	1611	1620#	1628										
BT026	002440	1625	1633#	1641										
BT027	002460	1638	1647#	1655										
BT030	002500	1652	1661#	1669										
BT031	002520	1666	1674#	1682										
BT032	002540	1679	1688#	1698	1704									
BT033	002600	1701	1710#	1720	1726									
BT034	002640	1723	1731#	1741	1747									
BT035	002700	1744	1752#	1761										
BT036	002720	1758	1766#	1775										
BT037	002740	1772	1780#	1789										
BT040	002760	1786	1795#	1806										
BT041	003006	1803	1811#	1822	1828									
BT042	003044	1825	1834#	1844										
BT043	003064	1841	1850#	1861										
BT044	003106	1858	1867#											
BT045	003134	1874	1877	1882#	1892									
BT046	003162	1889	1898#	1920	1926									
CATERR	063252	15840	15842#	15856#	15908	15910#	15917#	16388#						
CCNSC0	064572	16801#												
CCNSC1	064576	16802#												
CCNSC2	064602	16803#												
CCNST0	064616	16806#												
CCNST1	064622	16807#												
CCNST2	064612	16805#												
CCR =	177746	1224#	15229#	15231#	15281#	15295#	15297#	15412#	15413	15452#	15456#	15458#	15462#	15477#
CCST20	064606	16804#												
CITST	013412	4613	4625#											
CLCUA	064552	16797#												
CLFGIN	064556	16798#												
CLJAM	064536	16794#												
CLPBA	064546	16796#												
CLSERV	064542	16795#												
CLTAG	064566	16800#												

```
CLWORM  064562          16799#
CNSCOR  064234          16646#
CNSCTL  064442          16742#
CNSC0   064360          16692#
CNSC1   064362          16693#
CNSC2   064364          16694#
CNSSW   064232          16645#
CNSTSW  064230          16644#
CNST0   064370          16696#
CNST1   064434          16733#
CNST2   064432          16732#
COUNT   064455          16753#
CPUERR= 177766           1223#  15177  15343  15373  15421  15596  15625  15633
CR    = 000015            734#  16193  16203  16867  16871
CRLF  = 000200            735#  16164  16203
CSP1    064342          16684#
CSP2    064370          16696#
CST200  064366          16695#
CUR     064447          16747#
DBTA    064630           2774   3320   3332   3370   3382   3570   3578   9633   9653   9672   9683   9696   9707
                         9721   9742   9769   9796   9823   9850   9872  11023  11043  11064  11086  11108  11130
                        11152  11174  11260  16402  16811#
DBTB    064634           3345   3357   3395   3407   3446   3498   3659   3699  16403  16814#
DCS0    064460          16756#
DCS1    064462          16758#
DDISP = 177570            741#   906
DF15    067766           1045  17123#
DF17    067770           1072   1078  17124#
DHEIS1  067232           1201   1208  17054#
DHI1    065307           1019  16878#
DHI5    067314           1043  17063#
DHI7    067346           1070   1076  17068#
DHI2    065061            984  16849#
DHI23   067403           1025   1031   1037   1082   1117   1124   1145   1180  17073#
DHI24   067410           1089   1138   1152   1166   1173  17074#
DHI25   067433           1159  17078#
DHI26   067447           1103  17081#
DHI27   067531           1110  17090#
DHI32   067564           1131   1194  17095#
DHI4    065072            994  16851#
DHI42   067600           1187  17097#
DHI44   067617           1050   1057   1064   1096  17100#
DHI46   067273           1215  17060#
DISPLA  001042            906#  15990*  16033*
DISPRE  000174            841#
DPTRP = 000001           1226#  15231  15452  15462  15477
DREG    001452          16750#
DS    = 177570            740#   905
DTEIS1  067744           1202   1209  17118#
DTI1    067774            980  17126#
DTI10   067774           1015  17129#
DTI11   070034           1020  17137#
DTI15   067640           1044  17104#
DTI2    067774            985  17127#
DTI21   067650           1071  17106#
DTI22   067662           1077  17108#
```

```
DT23    067674          1026    1032    1038    1083    1118    1125    1146    1181   17110#
DT24    067700          1051    1058    1065    1090    1097    1139    1153    1167    1174   17111#
DT25    067710          1132    1160    1188    1195   17113#
DT26    067716          1104   17114#
DT27    067732          1111   17116#
DT3     070016           990   17134#
DT4     067774           995   17128#
DT46    067756          1216   17120#
DT5     070002          1000   17132#
DT6     070000          1005   17131#
DT7     067776          1010   17130#
DWTA    063322          2326    2328    2529    2575    2579    2598    2602    2647    2701    2719    2750    2810    2836
                        2855    2884    2917    3170    3171    3182    3188    3207    3208    3209    3219    3225    3244
                        3245    3246    3256    3262    3281    3282    3293    3299    3522    3534    3544    3556    6686
                        7635    7659    7683    7708    7732    7776    7797    7818    7839    7868    7897    7926    7955
                        7984    8013    8042    8071    8093    8115    8137    8634    8897    8929    8960    8992    9025
                        9062    9099    9136    9173    9210    9251    9288    9325    9357    9389    9421   10509   10536
                       10591   10618   10645   10671   10698   10753   10782   16400   16411#
DWTB    064032          2551    2553    2907    3420    3472    3606    3614    3641    3677    8607   16401   16593#
EA017   002166          1495    1496    1497    1500#
EA020   002242          1529#
EA021   002272          1551#
EA023   002360          1587    1588    1589    1592#
EA032   002564          1697#
EA033   002624          1719#
EA034   002664          1740#
EA041   003030          1821#
EA046   003230          1919#
EMEIS1  067075          1200   17037#
EMEIS2  067126          1207   17042#
EMTVEC= 000030           829#
EM1     064640           978   16820#
EM10    064724          1013   16832#
EM11    065271          1018   16875#
EM12    065345          1024   16884#
EM13    065404          1030   16890#
EM14    065436          1036   16895#
EM15    065456          1042   16898#
EM16    067045          1049   17032#
EM17    065521          1095   16904#
EM2     064640           983   16821#
EM21    065545          1069   16908#
EM22    065570          1075   16912#
EM23    065617          1081   16916#
EM24    065636          1088   16919#
EM26    065715          1063    1102   16927#
EM27    065742          1109   16931#
EM3     065013           988   16842#
EM30    066012          1116   16938#
EM31    066615          1123   17006#
EM32    066124          1130   16951#
EM33    066155          1137   16956#
EM34    066233          1144   16964#
EM35    066270          1151   16969#
EM36    066356          1158   16978#
EM37    066433          1165   16986#
```

EMM	064640	993	16822#					
EMM0	066526	1172	16996#					
EMM1	066055	1179	16944#					
EMM2	066642	1186	17010#					
EMM3	066730	1193	17019#					
EMM5	067016	1056	17028#					
EMM6	067154	1214	17046#					
EM5	064666	998	16827#					
EM6	064660	1003	16826#					
EM7	064650	1008	16824#					
EOP1	065103	15706	16853#					
EOP2	065121	15709	16856#					
E[illegible]D	062040	4427	16065#					
E[illegible]FLG	063240	4429#	4434#	16065#	16383#			
ERRVEC=	000004	822#	14908#	14921#	15955	15956#	15958#	15961#
EX002	001640	1291#						
E001	001632	1281#						
E003	001656	1301	1302	1303	1306#			
E004	001674	1316	1317	1318	1321#			
E005	001712	1331	1332	1333	1336#			
E006	001724	1350#						
E007	001740	1364#						
E010	001756	1377	1380#					
E011	001776	1393	1396#					
E012	002016	1411#						
E013	002042	1428#						
E014	002064	1443#						
E016	002136	1476	1477	1478	1481#			
E022	002330	1569	1570	1571	1574#			
E024	002414	1613#						
E025	002434	1624	1627#					
E026	002454	1637	1640#					
E027	002474	1654#						
E030	002514	1668#						
E031	002534	1681#						
E035	002714	1760#						
E036	002734	1774#						
E037	002754	1788#						
E040	003002	1805#						
E042	003060	1843#						
E043	003102	1860#						
E044	003130	1876#						
E045	003156	1891#						
E1015A	002102	1457#						
E2002	001644	1295#						
E2015	002112	1464#						
E2017	002206	1509#						
E2020	002252	1535#						
E2021	002304	1558#						
E2023	002370	1598#						
E2032	002574	1703#						
E2033	002634	1725#						
E2034	002674	1746#						
E2041	003040	1827#						
E2046	003242	1925#						
FAC0.0	064324	16676#						

```
FAC0.1  064326          16677#
FAC0.2  064330          16678#
FAC0.4  064332          16679#
FAC0.5  064334          16680#
FAC1.0  064236          16647#
FAC1.1  064240          16648#
FAC1.2  064242          16649#
FAC1.3  064244          16650#
FAC1.4  064246          16651#
FAC1.5  064250          16652#
FAC2.0  064272          16663#
FAC2.1  064274          16664#
FAC2.2  064276          16665#
FAC2.3  064300          16666#
FAC2.4  064302          16667#
FAC2.5  064304          16668#
FAC3.0  064202          16633#
FAC3.1  064204          16634#
FAC3.2  064206          16635#
FAC3.3  064210          16636#
FAC3.4  064212          16637#
FAC3.5  064214          16638#
FDST0   064340          16682#
FDST1   064254          16654#
FDST2   064310          16670#
FDST3   064220          16640#
FEA     064336          16681#
FLAG    064450          16748#
FPSHI   064252          16653#
GNS   = ****** U          840   16371   16372   16373   16374
GNHAM   064226          16643#
HI    = 000200           1229#  15490   15500
HT    = 000011            732#  16162   16203   16826   16827   16832   16842   16849   16851   16856   16878   17054   17060
                        17063   17068   17074   17078   17081   17090   17095   17097   17100
IBUF    063266           4585    4606   16398#
IDENT1  065141           4642   16859#
INCRSV  061202          15841   15856#
INIT    003262           1934#  15722   15855
INSTAB  063636          13862   16530#
IOTVEC= 000020            827#
I002    001642           1287    1293#
I004    001664           1316#
I005    001702           1331#
I006    001720           1346#
I007    001734           1360#
I010    001750           1375#
I011    001766           1390#
I012    002006           1406#
I013    002030           1422#
I014    002060           1439#
I015    002076           1453#
I016    002124           1474#
I017    002154           1493#
I020    002236           1525#
I021    002266           1547#
I022    002312           1567#
```

```
I023    002346          1585#
I025    002404          1609#
I026    002422          1622#
I028    002440          1635#
I027    002466          1650#
I030    002506          1664#
I031    002526          1677#
I032    002560          1693#
I033    002620          1715#
I034    002660          1736#
I035    002710          1756#
I036    002730          1770#
I037    002750          1784#
I040    002772          1800#
I041    003020          1816#
I042    003052          1838#
I043    003074          1855#
I044    003120          1871#
I045    003146          1886#
I046    003212          1910#
JAM     064444         16744#
JMPS    041420         11484   11499   11505#
JMPSA   041466         11514   11531#
LCOTA   064436         16738#
LCUA    064350         16688#
LF    = 000012           733#  16197   16203   16867   16871
LFGIN   064352         16689#
LJAM    064342         16685#
LKCSR = 177546          1268#  12309   12332   12350   12369   12399   12612   12643   12674   12705   12736   12771   12836
                       12883
LO    = 000100          1228#  15490   15500
LPBA    064346         16687#
LSERV   064344         16686#
LTAG    064356         16691#
LWAAM   064354         16690#
MBUF0   063312          2047   2049*   2063   2065*   2124   2142   2162   2181   2198   2203   2215   2220   2238
                        2243   2261    2266   2278    2282   2304   2315   2356   2362*  2375   2380*  2410   2414*
                        2427   2429    2658   2660    2677   2751   2785   2787   2811   2812   2835   2841*  2854
                        2860*  2917*   2918   2937*   2938   2950   2965   2971   2989   2995   3007   3012*  3025
                        3030*  3100    3119   3120    3318   3343   3368   3393   3418   3433   3444   3459   3470
                        3473   3485    3496   3499    3511   3639   3645*  3657   3663*  3675   3681*  3697   3703*
                        4111   4115*   4128*  4132    4489*  4490*  4991   5011   5026   5031   5044   5061   5087
                        5157   5177    5191   5196    5209   5313   5340   5367   5394   5577   5604   5632   5660
                        5688   5716    5744   5756    5774   5804   5806   5817   5835   5837   5861   5863   5674
                        5892   5894    5918   5930    5948   6595   6623   6656   6684   6698   6716   6743   6770
                        6797   6824    6851   6878    6904   6931   6958   6985   7012   7039   7066   7093   7120
                        7147   7174    7201   7228    7385   7387   7394   7412   7414   7434   7442   7460   7481
                        7483   7490    7512   7514    7534   7542   7560   7685*  7691   7734*  7740   7757   7778*
                        7796   7817    7838   7855    7867   7884   7896   7925   7954   7957   7983   7986   8012
                        8041   8070    8073   8092    8095   8114   8136   8267   8296   8329   8358   8520   8549
                        8578   8687    8717   8748    8762   8774   8800   8802   8826   8852   8854   8873   8896
                        8928   8959    8991   9024    9040   9061   9077   9098   9135   9172   9174   9209   9211
                        9250   9287    9324   9326    9356   9358   9388   9420   9450   9471   9492   9513   9515
                        9534   9558    9719   9740    9755   9767   9794   9798   9821   9848   9852   9870   9892
                        9914   9936    9958   9961    9980   9983   10249  10278  10307  10336  10365  10394  10422
                       10451  10480   10751  10780   10809  10839  10869  10903  10933  10963  10993  11041  11062
```

```
                    11084   11106   11109   11128   11150   11153   11172   11194   11215   11218   11237   11258   11261
                    12891*  12913*  12916   13867   13946*  13952   14346   14377   14397   14414*  14449   14450*  14469
                    14486*  14521   14553*  16404   16407#
MBUF1   063316       2161    2180    2718    2723*   2936    3317    3342    3367    3392    7850*   7854    7879*   7883
                     7908*   7912    7937*   7941    7966*   7970    7995*   7999    8024*   8028    8053*   8057    8661
                     8689    8719    9556   10563   10725   10811   10841   10871   10905   10935   10965   10995   12892*
                    12922   12929*  13161   13171   13200   13210   13868   13912*  13918   16405   16409#
MD      064440      16740#
MED    = 076600      1219#  13805   14912   14925   14956   14961   14964   14968   15002   15015   15044   15047   15055
                    15089   15092   15094   15097   15108   15121   15124   15128   15140   15143   15145   15166   15175
                    15190   15197   15209   15236   15238   15240   15245   15251   15257   15263   15269   15302   15307
                    15341   15357   15371   15388   15419   15460   15466   15475   15487   15508   15519   15533   15541
                    15547   15578   15594   15616   15631   15642   15660   15675   15755
MEDHLT  055632      14927   14928   14929   14931#
MEDT1   055642      14948#
MEDT10  056206      15049#
MEDT11  056260      15085#  15147
MEDT3   056042      14999#
MEDT4   056100      15013#
MEDO    055612      14924#
MEDI    055514      14905#
MEMERR= 177744       1222#  15310   15314   15499
NOPS    064404      16711#
NUA     064456      16754#
OBUF    063256       4586    4607   16393#
ONCE    063254       1931*   1936    4638    4640*  16390#
ORTAB   063474      14191   14196   16472#
PABORT= 000200       1227#  15452
PA1500  064162      16608#
PA1716  064160      16607#
PBA     064446      16746#
PIRQ   = 177772       739#
PIRQVE= 000240        833#
PRIFLG  063236       1934    4405*   4414*  16120*  16382#
PRINA   062202       4407   16120#
PR0    = 000000       756#
PR1    = 000040       757#
PR2    = 000100       758#
PR3    = 000140       759#
PR4    = 000200       760#
PR5    = 000240       761#
PR6    = 000300       762#
PR7    = 000340       763#
PS     = 177776       736#    737
PSW    = 177776       737#   1436    1450    1471    1489    1500    1509    1518    1567*   1581    1605    1609    1622
                     1635    1688    1689*   1710    1711*   1731    1732*   1795    1811    2001*   2016*   2030    4230*
                     4238    4242*   4261*   4266    4270*   4284*   4292    4296*   4303*   4310*   4345*   4353    4357*
                     4364*   4371*   4384*   4389    4392*   4643*   4685    4691    4705    4713    4727    4733    4747
                     4755   12114   12121*  12129   12151   12158*  12166   12215   12218   12275   12278   12375*  12406*
                    12421*  12432*  12438   12450   12453*  12465*  12471   12483   12486*  12511*  12517*  12534   12541*
                    12568*  12574*  12591   12598*  12617*  12632*  12648*  12663*  12679*  12694*  12710*  12725*  12741*
                    12756*  12776*  12792*  12808*  12824*  12844*  12854*  12890*  12911*  12947*  14686*  14715   14716
                    14906*  14924*  14948*  15779*
PSWHOL  064050      16602#
PTRP1   057712      15463   15474#
PWRUP   061054      15769   15778#
```

```
PWRVEC= 000024            828#  15733#  15734#  15743#  15749#  15764#  15765#
RCSR  = 177560           1264#   4496    4582
RDBR  = 177562           1265#
RDFLAG= 000144           1240#  15045   15095   15122
RDLCUR= 000103           1248#  15109
RDLDAT= 000106           1254#  15264   15534
RDLFGI= 000104           1250#  15389   15548   15661   15676
RDLJAM= 000100           1242#  15176   15342   15372   15420   15595   15632
RDLPBA= 000102           1246#  15191   15258   15358   15509
RDLSER= 000101           1244#  15198   15488
RDLTAG= 000107           1256#  15270   15520
RDLWRA= 000105           1252#
RDWRAM= 000022           1238#  15090   15141
RES     064457          16755#
RESTAR  061176          15844   15853   15855#  15858   15912   15919
RESVEC= 000010            823#  14909#  14922#
REV     064453          16751#
RSBERT  061150          15847#  15915
RSERR   061122           4460   13842   14922   15840#
RSMSG   065243          15846   16871#
RSVFLG  063246           4448#   4454#  15837#  16386#
RSVTST  061114           4445   15837#
RT1A    064372          16697#
RT1B    064376          16699#
RT2A    064374          16698#
RT2B    064400          16700#
RVECT   064322          16675#
RZERO   064320          16674#
R0A     064422          16728#
R0B     064426          16730#
R1A     064166          16627#
R1B     064256          16657#
R2A     064170          16628#
R2B     064260          16658#
R3A     064172          16629#
R3B     064262          16659#
R4A     064174          16630#
R4B     064264          16660#
R5A     064176          16631#
R5B     064266          16661#
R6A     064200          16632#
R6B     064270          16662#
R7A     064424          15013   16729#
R7B     064430          15025   16731#
SCOFLG  063244           4321#   4328#  15999#  16385#
SCOPEA  061612           4322   15999#
SELTST  063242          15945#  15946#  15947   16384#
SERV    064445          16745#
SOBERR  043012          11979   11988#
SOB1    042764          11974#  12057
SOB2    043000          11969   11972   11982#
SOB3    043156          11974   12055#
SOB4    043160          12053   12057#
SOB5    042772          11978#  11982
SREG    064454          16752#
STACK = 001000            727#   1435    1922    1938   14907   15778
```

```
START   001630            844    1278#  15780
STGBLK  064056          16605#
STKLMT= 177774            738#
SWBO1 = 000071           1258#  15085   15103   15107
SWR     001040            905#   4635*   4636   15741   15757*  15843   15852   15911   15943   15945   15950   15968   15975
                        16038   16050   16053
SWREG   000176            842#
SW0   = 000001            791#
SW00  = 000001            781#    791
SW01  = 000002            780#    790
SW02  = 000004            779#    789
SW03  = 000010            778#    788
SW04  = 000020            777#    787
SW05  = 000040            776#    786
SW06  = 000100            775#    785
SW07  = 000200            774#    784
SW08  = 000400            773#    783
SW09  = 001000            772#    782
SW1   = 000002            790#
SW10  = 002000            771#  15943
SW11  = 004000            770#
SW12  = 010000            769#   4636   15843   15911
SW13  = 020000            768#
SW14  = 040000            767#
SW15  = 100000            766#
SW2   = 000004            789#
SW3   = 000010            788#
SW4   = 000020            787#
SW5   = 000040            786#
SW6   = 000100            785#
SW7   = 000200            784#
SW8   = 000400            783#
SW9   = 001000            782#
TABBEG  064052          16603#
TABEND  064054          16604#
TAG   = 000040           1230#  15490   15500
TBITVE= 000014            824#
TBL2    064166          14949   16624#
TBL3    064404          14999   16710#
TBL4    064422          16727#
TBL5    064436          15114   16736#
TBL6    064464          15115   16764#
TBL7    064536          15051   16792#
TBSER   061070          13850   15786#
TKVEC = 000060            831#
TLOC1   064046          15227   15242   15293   15300   15454   15464   15510   15514   15522   16601#
TLOC2   064164          16609#
TPVEC = 000064            832#
TRAPVE= 000034            830#
TRTVEC= 000014            825#
TSET    052422          13850#
TST0    003316           1946#   4644
TST1    003332           1951    1959#
TST10   003532           2054    2061#
TST100  007200           3232    3241#
TST101  007274           3269    3278#
```

TST102	007366	3306	3315#
TST103	007446	3333	3340#
TST104	007526	3358	3365#
TST105	007606	3383	3390#
TST106	007666	3408	3416#
TST107	007744	3434	3442#
TST11	003564	2070	2078#
TST110	010022	3460	3468#
TST111	010102	3486	3494#
TST112	010162	3512	3520#
TST113	010224	3535	3542#
TST114	010266	3557	3565#
TST115	010356	3601#	
TST116	010446	3637#	
TST117	010510	3648	3655#
TST12	003622	2092	2100#
TST120	010552	3666	3673#
TST121	010614	3684	3691#
TST122	010670	3706	3714#
TST123	010714	3722	3730#
TST124	010752	3750	3758#
TST125	010770	3764	3772#
TST126	011004	3778	3786#
TST127	011020	3792	3800#
TST13	003656	2114	2122#
TST130	011040	3807	3815#
TST131	011060	3822	3830#
TST132	011076	3837	3845#
TST133	011114	3851	3859#
TST134	011132	3866	3874#
TST135	011150	3881	3889#
TST136	011170	3896	3904#
TST137	011210	3911	3919#
TST14	003710	2132	2140#
TST140	011230	3926	3934#
TST141	011250	3941	3949#
TST142	011264	3955	3963#
TST143	011304	3970	3978#
TST144	011324	3985	3993#
TST145	011344	4000	4008#
TST146	011362	4015	4023#
TST147	011376	4029	4037#
TST15	003744	2150	2158#
TST150	011416	4044	4052#
TST151	011436	4059	4067#
TST152	011456	4074	4082#
TST153	011524	4100	4108#
TST154	011566	4118	4126#
TST155	011622	4135	4143#
TST156	011710	4171	4179#
TST157	011744	4192	4200#
TST16	004010	2169	2177#
TST160	012012	4216	4224#
TST161	012100	4247	4255#
TST162	012152	4268	4277#
TST163	012274	4308	4317#

```
TST164  012342          4338#
TST165  012464          4369    4378#
TST166  012540          4401#
TST167  012620          4415    4423#
TST17   004054          2188    2196#
TST170  012664          4435    4442#
TST171  012746          4466#
TST172  013016          4479    4487#
TST173  013116          4515#
TST174  013156          4525    4536#
TST175  013210          4545    4554#
TST176  013244          4563    4580#
TST177  013256          4652#
TST2    003360          1965    1973#
TST20   004110          2206    2213#
TST200  013600          4659    4666#
TST201  013620          4673    4680#
TST202  013660          4693    4700#
TST203  013722          4715    4722#
TST204  013760          4735    4742#
TST205  014020          4757    4764#
TST206  014042          4771    4778#
TST207  014064          4786    4793#
TST21   004154          2228    2236#
TST210  014106          4801    4808#
TST211  014130          4816    4823#
TST212  014154          4831    4838#
TST213  014176          4845    4852#
TST214  014220          4860    4867#
TST215  014242          4875    4882#
TST216  014264          4890    4897#
TST217  014332          4916    4923#
TST22   004214          2251    2259#
TST220  014374          4937    4944#
TST221  014442          4963    4970#
TST222  014474          4980    4987#
TST223  014622          5032    5039#
TST224  014656          5049    5056#
TST225  014726          5075    5082#
TST226  014764          5093    5100#
TST227  015034          5119    5126#
TST23   004250          2268    2276#
TST230  015104          5145    5152#
TST231  015234          5197    5204#
TST232  015310          5224    5231#
TST233  015354          5250    5257#
TST234  015424          5276    5283#
TST235  015472          5302    5309#
TST236  015542          5329    5336#
TST237  015616          5356    5363#
TST24   004310          2290    2298#
TST240  015672          5383    5390#
TST241  015744          5410    5417#
TST242  016014          5436    5443#
TST243  016062          5462    5469#
TST244  016132          5488    5495#
```

TST245	016200	5514	5521#
TST246	016250	5540	5547#
TST247	016320	5566	5573#
TST25	004460	2316	2324#
TST250	016376	5593	5600#
TST251	016452	5620	5628#
TST252	016530	5648	5656#
TST253	016604	5676	5684#
TST254	016662	5704	5712#
TST255	016740	5732	5740#
TST256	017020	5762	5770#
TST257	017072	5788	5796#
TST26	004434	2345	2354#
TST260	017170	5823	5831#
TST261	017244	5849	5857#
TST262	017330	5880	5888#
TST263	017404	5906	5914#
TST264	017464	5936	5944#
TST265	017534	5961	5969#
TST266	017600	5988	5995#
TST267	017646	6015	6022#
TST27	004470	2365	2373#
TST270	017714	6041	6048#
TST271	017760	6066	6073#
TST272	020030	6092	6099#
TST273	020076	6118	6125#
TST274	020144	6144	6151#
TST275	020214	6170	6177#
TST276	020262	6196	6203#
TST277	020332	6222	6229#
TST3	003364	1978	1985#
TST30	004526	2383	2391#
TST300	020376	6247	6254#
TST301	020444	6273	6280#
TST302	020514	6299	6306#
TST303	020560	6325	6332#
TST304	020630	6351	6358#
TST305	020700	6376	6383#
TST306	020744	6402	6409#
TST307	021014	6428	6435#
TST31	004552	2399	2407#
TST310	021062	6454	6461#
TST311	021132	6480	6487#
TST312	021200	6506	6513#
TST313	021250	6532	6539#
TST314	021320	6558	6565#
TST315	021366	6584	6591#
TST316	021440	6611	6619#
TST317	021516	6640	6648#
TST32	004606	2417	2425#
TST320	021604	6672	6680#
TST321	021666	6704	6712#
TST322	021744	6731	6739#
TST323	022020	6758	6766#
TST324	022074	6785	6793#
TST325	022152	6812	6820#

TST326	022226	6839	6847#
TST327	0[illegible]304	6[illegible]56	6874#
TST33	00[illegible]46	2436	2444#
TST330	022356	6892	6900#
TST331	0[illegible]432	6919	6927#
TST332	0[illegible]510	6946	6954#
TST333	0[illegible]562	6973	6981#
TST334	022640	7000	7008#
TST335	022716	7027	7035#
TST336	022770	7054	7062#
TST337	023046	7081	7089#
TST34	004702	2457	2465#
TST340	023122	7108	7116#
TST341	023200	7135	7143#
TST342	023254	7162	7170#
TST343	023332	7189	7197#
TST344	023410	7216	7224#
TST345	023464	7243	7251#
TST346	023534	7270	7277#
TST347	023604	7296	7303#
TST35	004736	2478	2485#
TST350	023654	7322	7329#
TST351	023724	7348	7355#
TST352	023774	7374	7381#
TST353	024054	7400	7408#
TST354	024124	7422	7430#
TST355	024200	7448	7456#
TST356	024244	7469	7477#
TST357	024324	7496	7504#
TST36	004772	2498	2506#
TST360	024406	7522	7530#
TST361	024462	7548	7556#
TST362	024526	7569	7577#
TST363	024630	7597	7604#
TST364	024654	7624	7631#
TST365	024722	7648	7655#
TST366	024770	7672	7679#
TST367	025044	7697	7704#
TST37	005026	2519	2527#
TST370	025112	7721	7728#
TST371	025166	7746	7753#
TST372	025226	7765	7772#
TST373	025272	7785	7792#
TST374	025336	7805	7813#
TST375	025402	7826	7834#
TST376	025470	7856	7863#
TST377	025556	7885	7892#
TST4	003402	1991	1999#
TST40	005066	2541	2549#
TST400	025646	7914	7921#
TST401	025736	7943	7950#
TST402	026024	7972	7979#
TST403	026112	8001	8008#
TST404	026202	8030	8037#
TST405	026272	8059	8066#
TST406	026344	8080	8088#

TST407	026416	8102	8110#
TST41	005132	2564	2573#
TST410	0[illegible]466	8124	8132#
TST411	0[illegible]536	8146	8154#
TST412	0[illegible]602	8173	8180#
TST413	0[illegible]672	8200	8207#
TST414	0[illegible]726	8227	8234#
TST415	027002	8254	8261#
TST416	027056	8282	8290#
TST417	027136	8311	8319#
TST42	005174	2587	2596#
TST420	027232	8344	8352#
TST421	027314	8373	8381#
TST422	027364	8401	8408#
TST423	027434	8428	8435#
TST424	027510	8455	8462#
TST425	027564	8482	8489#
TST426	027640	8509	8516#
TST427	027720	8537	8545#
TST43	005236	2610	2619#
TST430	030000	8566	8574#
TST431	030060	8594	8602#
TST432	030132	8622	8629#
TST433	030202	8649	8656#
TST434	030260	8676	8683#
TST435	030344	8705	8713#
TST436	030432	8736	8744#
TST437	030506	8763	8770#
TST44	005274	2640#	
TST440	030564	8789	8796#
TST441	030640	8815	8822#
TST442	030716	8841	8848#
TST443	030766	8861	8869#
TST444	031036	8882	8890#
TST445	031122	8913	8922#
TST446	031206	8945	8954#
TST447	031272	8977	8986#
TST45	005322	2648	2657#
TST450	031356	9009	9018#
TST451	031452	9046	9055#
TST452	031546	9083	9092#
TST453	031644	9120	9129#
TST454	031742	9157	9166#
TST455	032036	9194	9203#
TST456	032132	9231	9240#
TST457	032242	9272	9281#
TST46	005364	2667	2675#
TST460	032340	9309	9318#
TST461	032430	9341	9350#
TST462	032520	9373	9382#
TST463	032610	9405	9414#
TST464	032700	9437	9446#
TST465	032740	9459	9467#
TST466	033000	9480	9488#
TST467	033042	9501	9509#
TST47	005420	2685	2693#

TST470	033104	9522	9530#
TST471	033146	9543	9552#
TST472	033212	9565	9573#
TST473	033264	9593	9600#
TST474	033336	9621	9628#
TST475	033374	9641	9648#
TST476	033434	9660	9667#
TST477	033502	9684	9691#
TST5	003424	2006	2014#
TST50	005462	2708	2716#
TST500	033650	9708	9715#
TST501	033614	9728	9736#
TST502	033672	9756	9763#
TST503	033752	9782	9790#
TST504	034030	9809	9817#
TST505	034110	9836	9844#
TST506	034162	9858	9866#
TST507	034234	9880	9888#
TST51	005520	2725	2733#
TST510	034302	9902	9910#
TST511	034350	9924	9932#
TST512	034420	9946	9954#
TST513	034470	9968	9976#
TST514	034542	9990	9998#
TST515	034616	10018	10025#
TST516	034676	10049	10056#
TST517	034752	10076	10083#
TST52	005544	2748#	
TST520	035020	10103	10110#
TST521	035076	10130	10138#
TST522	035150	10158	10165#
TST523	035222	10185	10192#
TST524	035274	10212	10219#
TST525	035346	10238	10245#
TST526	035430	10266	10274#
TST527	035504	10295	10303#
TST53	005600	2758	2766#
TST530	035566	10324	10332#
TST531	035642	10353	10361#
TST532	035724	10382	10390#
TST533	036006	10411	10418#
TST534	036070	10439	10447#
TST535	036150	10468	10476#
TST536	036230	10496	10504#
TST537	036304	10524	10531#
TST54	005632	2775	2783#
TST540	036354	10551	10558#
TST541	036434	10579	10586#
TST542	036504	10606	10613#
TST543	036556	10633	10640#
TST544	036630	10660	10666#
TST545	036702	10686	10693#
TST546	036754	10713	10720#
TST547	037032	10740	10747#
TST55	005700	2800	2808#
TST550	037114	10768	10776#

F10


```
TST551  037172          10797   10805#
TST552  037260          10827   10835#
TST553  037340          10857   10865#
TST554  037426          10887   10895#
TST555  037526          10921   10929#
TST556  037612          10951   10959#
TST557  037676          10981   10989#
TST56   005750           2825    2833#
TST560  037762          11010   11018#
TST561  040020          11030   11037#
TST562  040064          11050   11058#
TST563  040132          11072   11080#
TST564  040202          11094   11102#
TST565  040252          11116   11124#
TST566  040322          11138   11146#
TST567  040374          11160   11168#
TST57   006010           2844    2852#
TST570  040446          11182   11190#
TST571  040510          11203   11211#
TST572  040562          11225   11233#
TST573  040622          11246   11254#
TST574  040672          11268   11276#
TST575  040732          11286   11291   11298#
TST576  040772          11308   11313   11320#
TST577  041042          11330   11340   11347#
TST6    003444           2020    2028#
TST60   006062           2869    2876#
TST600  041102          11357   11362   11369#
TST601  041160          11379   11389   11392   11401#
TST602  041226          11411   11416   11419   11427#
TST603  041304          11437   11441   11451   11458#
TST604  041344          11468   11473   11480#
TST605  041422          11490   11500   11503   11510#
TST606  041470          11520   11525   11528   11536#
TST607  041536          11546   11551   11554   11562#
TST61   006124           2891    2899#
TST610  041604          11572   11577   11580   11588#
TST611  041660          11598   11601   11606   11609   11618#
TST612  041734          11628   11631   11636   11639   11648#
TST613  042012          11664   11676#
TST614  042100          11707#
TST615  042162          11736#
TST616  042232          11761#
TST617  042310          11777   11790#
TST62   006156           2908    2915#
TST620  042374          11806   11823#
TST621  042456          11842   11855#
TST622  042542          11871   11888#
TST623  042622          11904   11916#
TST624  042710          11933   11950#
TST625  042740          11958   11966#
TST626  043014          11984   11987   11993#
TST627  043050          12005   12012#
TST63   006216           2925    2933#
TST630  043104          12024   12031#
TST631  043140          12043   12050#
```

TST632	043174	12062	12069#			
TST633	043276	12099	12110#			
TST634	043402	12136	12147#			
TST635	043510	12173	12184#			
TST636	043666	12200	12208	12220	12229	12244#
TST637	044044	12260	12268	12280	12289	12304#
TST64	006274	2951	2959#			
TST640	044126	12328#				
TST641	044162	12338	12346#			
TST642	044220	12356	12364#			
TST643	044332	12394#				
TST644	044450	12426#				
TST645	044566	12459#				
TST646	044702	12492#				
TST647	045106	12549#				
TST65	006352	2979	2987#			
TST650	045310	12606#				
TST651	045426	12637#				
TST652	045544	12668#				
TST653	045662	12699#				
TST654	046000	12730#				
TST655	046116	12761#				
TST656	046250	12797#				
TST657	046370	12832#				
TST66	006406	2997	3005#			
TST660	046530	12877#				
TST661	047030	12960#				
TST662	047076	12980#				
TST663	047232	13020#				
TST664	047346	13058#				
TST665	047430	13083#				
TST666	047512	13108#				
TST667	047574	13133#				
TST67	006444	3015	3023#			
TST670	047660	13157#				
TST671	047766	13188	13196#			
TST672	050074	13226	13234#			
TST673	050234	13265	13286#			
TST674	050416	13334	13346#			
TST675	050600	13394	13406#			
TST676	050674	13436#				
TST677	051000	13468#				
TST7	003476	2038	2045#			
TST70	006504	3033	3041#			
TST700	051104	13500#				
TST701	051164	13523#				
TST702	051244	13546#				
TST703	051324	13568#				
TST704	051424	13601#				
TST705	051530	13619	13636#			
TST706	051636	13654	13671#			
TST707	051744	13689	13706#			
TST71	006534	3050	3058#			
TST710	052032	13732#				
TST711	052120	13759#				
TST712	052206	13786#				

```
TST713  052304          13816#
TST714  052436          13859#
TST715  052526          13886#
TST716  052556          13901#
TST717  052644          13926#
TST72   006604           3082#
TST720  052700          13942#
TST721  052742          13959#
TST722  053010          14008#
TST723  053074          14068#
TST724  053160          14128#
TST725  053244          14188#
TST726  053330          14234#
TST727  053404          14252   14278#
TST73   006634           3090    3098#
TST730  053460          14296   14308#
TST731  053544          14331   14338#
TST732  053634          14362   14369#
TST733  053732          14398   14405#
TST734  054036          14434   14441#
TST735  054142          14470   14477#
TST736  054246          14506   14513#
TST737  054340          14537   14544#
TST74   006672           3108    3116#
TST740  054422          14561   14568#
TST741  054506          14590   14597#
TST742  054574          14619   14626#
TST743  054656          14648   14655#
TST744  054730          14670   14681#
TST745  055046          14713   14723#
TST746  055116          14741   14747#
TST747  055166          14766   14772#
TST75   006734           3127    3135#
TST750  056236          14790   14796#
TST751  056324          14818   14824#
TST752  056406          14847   14853#
TST753  056506          14879   14902#
TST754  055634          14945#
TST755  056034          14996#
TST756  056164          15040#
TST757  056252          15082#
TST76   007012           3159    3167#
TST760  056554          15160#
TST761  056732          15223#
TST762  057142          15290#
TST763  057274          15331#
TST764  057502          15407#
TST765  057600          15448#
TST766  060160          15573#
TST767  060402          15648#
TST77   007104           3195    3204#
TYPE  = 104401           4410    4641   15706  15709  15712  15766  15845  15849  15913  16041  16076  16093  16095
                        16098   16100  16104  16111  16167  16264  16371#
TYPOC = 104402          15708   15711  15848  16084  16108  16372#
TYPON = 104404          16374#
TYPOS = 104403          16373#
```

```
UBREAK= 177770          1221#   1939*  14905*  15085*  15119*  15149*
UCNSCT  064474         16770#
UCOUNT  064522         16781#
UCUA    064506         16775#
UDCS0   064530         16784#
UDCS1   064534         16786#
UDREG   064514         16778#
UFLAG   064510         16776#
UINIT   064532         16785#
UJAM    064500         16772#
ULCDTA  064464         16766#
UM    = 140000          1220#
UMD     064470         16768#
UMUA    064524         16782#
UPBA    064504         16774#
URES    064526         16783#
UREV    064516         16779#
UR6A    064216         16639#
UR6B    064306         16669#
USERV   064502         16773#
USREG   064520         16780#
VADR    064156         16606#
WCMSSW= 000226          1237#  15756
WCSADR  064316         16673#
WCSA.0  064222         16641#
WCSA.1  064224         16642#
WCSB.0  064312         16671#
WCSB.1  064314         16672#
WRFLAG= 000344          1241#  15048   15098   15125   15144
WRLCUA= 000303          1249#
WRLDAT= 000306          1255#  15239
WRLFGI= 000304          1251#
WRLJAM= 000300          1243#
WRLPBA= 000302          1247#  15237
WRLSER= 000301          1245#
WRLTAG= 000307          1257#  15241
WRLWBA= 000305          1253#
WRWBAM= 000222          1239#  15093   15146   15167   15210   15461   15542   15579   15617   15643
WWP   = 000100          1225#  15229   15295   15297   15456   15458
XCSR  = 177564          1266#   4517    4538    4556   12431   12454*  12464   12487*  12498   12555   12803   12884
XDBR  = 177566          1267#
$APTHD  000700           855     861#
$ASTAT= ****** U       16316   16331
$ATYC   062744         16287   16289#
$ATY1   062720         16285#
$ATY3   062726         16152   16286#
$ATY4   062736         16046   16288#
$AUTOB  001034           902#
$BOADR  001022           897#
$BOOAT  001026           899#
$CHARC  062466         16169*  16179*  16186   16195*  16200#
$CKSWR= ****** U       16377
$CMTAG  001000           885#
$CM1  = 000006           917#    918#    919#    920#    921#    922#    923#
$CM2  = 000014           917#    918#    919#    920#    921#    922#    923#
$CM3  = 000006           915#    917
```

```
$CMM  = 000005          923#   924#   925#   926#   927#   928#
$CPUOP  001146          952#
$CRLF   001115          931#  15712  15850  16041  16064  16076  16095  16100  16104  16168  16203
$DEVCT  001130          943#
$DOAGN  060654        15702  15714  15720#
$ENDAD  060644          874  15716#
$ENDCT  060600        15704#
$ENULL  060660        15723#
$ENV    001140          948#  16043  16147  16294  16318
$ENVM   001141          949#   4633  16149  16154  16296
$EOP    060550        15695#
$EOPCT  060572        15701#  15705
$ERFLG  001003          898#  15928  15964  15966  15972*  15998  16031*  16064
$ERMAX  001015          894#  15966  15989*  15998
$ERROR  061620         4627  15655  16018#
$ERRPC  001016          895#  16035*  16036*  16037  16064  16082  17104  17106  17108  17110  17111  17113  17114
                      17116  17120  17132  17134  17137
$ERRTB  001150          975#  16090
$ERRTY  062046        16040  16075#
$ERTTL  001012          892#   1932*  15710  15851*  16034*  16064
$ESCAP  001112          929#  15939*  15941*  15988*  16056  16058  16064
$ETABL  001140          947#
$ETEND  001150          867    959#
$FATAL  001122          940#  16322*
$FFLG   063164        16285*  16288*  16316  16325*  16333#
$FILLC  001056          913#  16172  16203
$FILLS  001055          912#  16203
$GDAOR  001020          896#
$GDOAT  001024          898#
$GET42  060634        15713#
$GTSWR=  ******  U      16376
$HD   = 000000          712
$HIBTS  000700          862#
$ICNT   001004          889#  15979*  15980  15982*  15997
$ILLUP  061036        15733  15749  15771#
$INTAG  001035          903#
$ITEMB  001014          893#  16037*  16045  16064  16079
$LF     001116          932#  16064  16203
$LFLG   063163        16326*  16332#
$LPAOR  001006          890#   4644*  15970*  15986*  15991  15997
$LPERR  001010          891#  11653*  11681*  11716*  11741*  11766*  11795*  11828*  11860*  11893*  11921*  12074*  12116*
                     12153*  12310*  12371*  12400*  12497*  12554*  12611*  12642*  12673*  12704*  12735*  12770*  12802*
                     12838*  12882*  12965*  13138*  13505*  13528*  13551*  13573*  13713*  13728*  13739*  13754*  13766*
                     13781*  13793*  13811*  13823*  13841*  13865*  13881*  13910*  13964*  14012*  14034*  14072*  14094*
                     14132*  14154*  14192*  14214*  14240*  14259*  14284*  14303*  15116*  15147*  15970  15987*  15997
                     16055
$MAIL   001120          863    867    938#  15985  16043  16147
$MBAOR  000702          863#
$MFLG   063162        16286*  16292  16327*  16331#
$MSGAD  001134          945#  16302*  16305
$MSGLG  001136          946#  16307*
$MSGTY  001120          939#  16300  16308*  16320  16324*
$MXCNT  061610        15983  15997#
$NULL   001054          911#  16174  16203
$NWTST=  000001        1943#  1956#  1970#  1982#  1996#  2011#  2025#  2042#  2058#  2075#  2097#  2119#  2137#
                      2155#  2174#  2193#  2210#  2233#  2256#  2273#  2295#  2321#  2351#  2370#  2388#  2404#
```

```
                2422#  2441#  2462#  2482#  2503#  2524#  2546#  2570#  2593#  2616#  2637#  2653#  2672#
                2690#  2713#  2730#  2745#  2763#  2780#  2805#  2830#  2849#  2873#  2896#  2912#  2930#
                2956#  2984#  3002#  3020#  3043#  3055#  3079#  3095#  3113#  3132#  3164#  3201#  3238#
                3275#  3312#  3337#  3352#  3387#  3413#  3439#  3465#  3491#  3517#  3539#  3562#  3598#
                3634#  3652#  3670#  3688#  3711#  3727#  3755#  3769#  3783#  3797#  3812#  3827#  3842#
                3856#  3871#  3905#  3901#  3916#  3931#  3946#  3960#  3975#  3990#  4005#  4020#  4034#
                4049#  4064#  4079#  4105#  4123#  4140#  4176#  4197#  4221#  4252#  4274#  4314#  4335#
                4375#  4398#  4420#  4439#  4463#  4484#  4512#  4533#  4551#  4569#  4571    4649#  4663#
                4677#  4697#  4719#  4739#  4761#  4775#  4790#  4815#  4830#  4835#  4849#  4864#  4879#
                4894#  4920#  4941#  4967#  4984#  5036#  5053#  5079#  5097#  5123#  5149#  5201#  5228#
                5254#  5280#  5306#  5333#  5360#  5387#  5414#  5440#  5466#  5492#  5518#  5544#  5570#
                5597#  5625#  5653#  5681#  5709#  5737#  5767#  5793#  5819#  5864#  5885#  5911#  5941#
                5966#  5992#  6019#  6045#  6070#  6096#  6122#  6148#  6174#  6200#  6226#  6251#  6277#
                6303#  6329#  6355#  6380#  6406#  6432#  6458#  6484#  6510#  6536#  6562#  6588#  6616#
                6645#  6677#  6709#  6736#  6763#  6790#  6817#  6844#  6871#  6897#  6924#  6951#  6978#
                7005#  7032#  7059#  7086#  7113#  7140#  7167#  7194#  7221#  7248#  7274#  7300#  7326#
                7352#  7378#  7405#  7427#  7453#  7474#  7501#  7527#  7553#  7574#  7601#  7628#  7652#
                7676#  7701#  7725#  7750#  7769#  7789#  7810#  7831#  7860#  7889#  7918#  7947#  7976#
                8005#  8034#  8063#  8085#  8107#  8129#  8151#  8177#  8204#  8231#  8258#  8287#  8316#
                8349#  8378#  8405#  8432#  8459#  8486#  8513#  8542#  8571#  8599#  8626#  8653#  8680#
                8710#  8741#  8767#  8793#  8819#  8845#  8866#  8887#  8919#  8951#  8983#  9015#  9052#
                9089#  9126#  9163#  9200#  9237#  9278#  9315#  9347#  9379#  9411#  9443#  9464#  9485#
                9506#  9527#  9549#  9570#  9597#  9625#  9645#  9664#  9688#  9712#  9733#  9760#  9787#
                9814#  9841#  9863#  9885#  9907#  9929#  9951#  9973#  9995#  10022#  10053#  10080#  10107#
               10135#  10162#  10189#  10216#  10242#  10271#  10300#  10329#  10358#  10387#  10415#  10444#  10473#
               10501#  10528#  10555#  10583#  10610#  10637#  10663#  10690#  10717#  10744#  10773#  10802#  10832#
               10862#  10892#  10926#  10956#  10986#  11015#  11034#  11055#  11077#  11099#  11121#  11143#  11165#
               11187#  11208#  11230#  11251#  11273#  11295#  11317#  11344#  11366#  11398#  11424#  11455#  11477#
               11507#  11533#  11559#  11585#  11615#  11645#  11673#  11704#  11733#  11758#  11787#  11820#  11852#
               11885#  11913#  11947#  11963#  11990#  12009#  12028#  12047#  12066#  12107#  12144#  12181#  12241#
               12301#  12325#  12343#  12361#  12391#  12423#  12456#  12489#  12546#  12603#  12634#  12665#  12696#
               12727#  12758#  12794#  12826#  12828    12870#  12872    12957#  12977#  13017#  13055#  13080#  13105#
               13130#  13154#  13193#  13231#  13283#  13343#  13403#  13433#  13465#  13497#  13520#  13543#  13565#
               13596#  13598    13633#  13668#  13703#  13729#  13756#  13783#  13813#  13853#  13855    13883#  13898#
               13923#  13939#  13956#  13976#  13978    14036#  14038    14096#  14098    14156#  14158    14216#  14218
               14261#  14263    14305#  14335#  14366#  14402#  14438#  14474#  14510#  14541#  14565#  14594#  14623#
               14652#  14674#  14676    14720#  14744#  14769#  14793#  14821#  14850#  14895#  14897    14933#  14935
               14984#  14986    15030#  15032    15065#  15067    15152#  15154    15213#  15215    15283#  15285    15320#
               15322    15397#  15399    15436#  15438    15558#  15560    15645#
SOCNT   062714 16236#  16265#  16278#
SOMODE  062716 16231#  16235#  16240    16243#  16254#  16280#
SOVER   061560 15948    15951    15971    15981    15990#
SPASS   001126   942#   1933#  15698#  15699#  15707    15723    15977    15998
SPASTM  000706   865#
SPOWER  061044 15767    15774#
SPWRDO  061032 15769#
SPWRON  060664  4631    15733#  15764
SPWRMG  061026 15767#
SPWRUP  060736 15743    15749#
SQUES   001114   930#   16064    16203
SROCHR= ****** U 16377
SRODEC= ****** U 16377
SROLIN= ****** U 16377
SROOCT= ****** U 16377
SREGAD  001060   915#   16020    17118    17132    17134
SREGO   001062   917#   15453#  15464#  15481    15551#  17106    17111    17113    17114    17120    17132    17134    17137
```

```
$REG1   001064            918#  17111   17114   17116   17118   17132   17134
$REG2   001066            919#  17114   17116   17118   17131
$REG3   001070            920#  17114   17116   17118   17130   17134
$REG4   001072            921#  17118   17129   17134
$REG5   001074            922#   1940*  12196*  12204*  12216*  12234*  12256*  12264*  12276*  12294*  12995*  13003*  13010*
                        13034*  13042*  13049*  13073*  13098*  13123*  13299*  13309*  13319*  13329*  13359*  13369*  13379*
                        13389*  13417*  13423*  13427*  13449*  13458*  13481*  13490*  16060*  17132
$RTNAD  060656          15722#
$R2A    = ****** U      16377
$SAVRE= ****** U        16377
$SAVR6  061042          15742*  15750   15751*  15752*  15773#
$SCOPE  061260           4625   15670   15936#
$SETUP= 000000          15697   15937   16019   16053   16060
$SVLAD  061524          15959   15984#
$SVPC   = 000714          872#    877
$SWR    = 165000          702#    712     716     717     718     719     720     721     722     723     928     929     930
                         1948    1961    1975    1987    2001    2016    2030    2047    2063    2080    2102    2124    2142
                         2160    2179    2198    2215    2238    2261    2278    2300    2326    2356    2375    2393    2409
                         2427    2446    2467    2487    2508    2529    2551    2575    2598    2621    2642    2658    2677
                         2695    2718    2735    2750    2768    2785    2810    2835    2854    2878    2901    2917    2935
                         2961    2989    3007    3025    3043    3060    3084    3100    3118    3137    3169    3206    3243
                         3280    3317    3342    3367    3392    3418    3444    3470    3496    3522    3544    3567    3603
                         3639    3657    3675    3693    3716    3732    3760    3774    3788    3802    3817    3832    3847
                         3861    3876    3891    3906    3921    3936    3951    3965    3980    3995    4010    4025    4039
                         4054    4069    4084    4110    4128    4145    4181    4202    4226    4257    4279    4319    4340
                         4380    4403    4425    4444    4468    4489    4517    4538    4556    4582    4656    4670    4684
                         4704    4726    4746    4768    4782    4797    4812    4827    4842    4856    4871    4886    4901
                         4927    4948    4974    4991    5043    5060    5086    5104    5130    5156    5208    5235    5261
                         5287    5313    5340    5367    5394    5421    5447    5473    5499    5525    5551    5577    5604
                         5632    5660    5688    5716    5744    5774    5800    5835    5861    5892    5918    5948    5973
                         5999    6026    6052    6077    6103    6129    6155    6181    6207    6233    6258    6284    6310
                         6336    6362    6387    6413    6439    6465    6491    6517    6543    6569    6595    6623    6652
                         6684    6716    6743    6770    6797    6824    6851    6878    6904    6931    6958    6985    7012
                         7039    7066    7093    7120    7147    7174    7201    7228    7255    7281    7307    7333    7359
                         7385    7412    7434    7460    7481    7508    7534    7560    7581    7608    7635    7659    7683
                         7708    7732    7757    7776    7796    7817    7838    7867    7896    7925    7954    7983    8012
                         8041    8070    8092    8114    8136    8158    8184    8211    8238    8265    8294    8323    8356
                         8385    8412    8439    8466    8493    8520    8549    8578    8606    8633    8660    8687    8717
                         8748    8774    8800    8826    8852    8873    8894    8926    8958    8990    9022    9059    9096
                         9133    9170    9207    9244    9285    9322    9354    9376    9418    9450    9471    9492    9513
                         9534    9556    9577    9604    9632    9652    9671    9695    9719    9740    9767    9794    9821
                         9848    9870    9892    9914    9936    9958    9980   10002   10029   10060   10087   10114   10142
                        10169   10196   10223   10249   10278   10307   10336   10365   10394   10422   10451   10480   10508
                        10535   10562   10590   10617   10644   10670   10697   10724   10751   10780   10809   10839   10869
                        10899   10933   10963   10993   11022   11041   11062   11084   11106   11128   11150   11172   11194
                        11215   11237   11258   11280   11302   11324   11351   11373   11405   11431   11462   11484   11514
                        11540   11566   11592   11622   11652   11680   11711   11740   11765   11794   11827   11859   11892
                        11920   11954   11969   11997   12016   12035   12053   12073   12114   12151   12188   12248   12308
                        12332   12350   12368   12398   12430   12463   12496   12553   12610   12641   12672   12703   12734
                        12765   12801   12836   12881   12964   12984   13024   13062   13087   13112   13137   13161   13200
                        13238   13290   13350   13410   13440   13472   13504   13527   13550   13572   13605   13640   13675
                        13709   13735   13762   13789   13819   13861   13890   13905   13930   13946   13963   14011   14071
                        14131   14191   14237   14281   14311   14341   14372   14408   14444   14480   14516   14547   14571
                        14600   14629   14658   14685   14727   14751   14776   14799   14827   14856   14905   14948   14999
                        15043   15085   15163   15226   15293   15334   15410   15451   15576   15651   15692   15697   15715
                        15721   15723   15770   15929   15930   15931   15932   15933   15950   15962   15964   15965   15966
```

		15973	15974	15975	15987	15990	15997	16011	16012	16013	16014	16034	16038	16050
		16053	16064											
$SWREG	001142	950#	4635											
$S.RMK=	000000	15933												
$TESTN	001124	941#	4646#	15937	15985#	15995#	17137							
$TIMES	001110	928#	4645#	12430#	12463#	15697#	15973#	15980	15983#	15997				
$TKB	001046	908#												
$TKS	001044	907#												
$TMP0	001076	923#	14311#	14341#	14372#	14408#	14444#	14480#	14516#	14547#	14571#	14600#	14629#	14658#
		14799#	14827#	14860#	14958#	14967	15102#	15132#	17104	17120				
$TMP1	001100	924#	14959#	14972#	14978	15019#	15059#	15103#	15133#	17104	17106	17108		
$TMP2	001102	925#	14950#	14960	14970	14977#	14979	15020#	15060#	17106	17108			
$TMP3	001104	926#	14966#	14970	15021#	17108								
$TMP4	001106	927#	14955#	14973	14976#									
$TN =	000770	702#	712	1943	1747	1948#	1951	1956	1960	1961#	1965	1970	1974	1975#
		1978	1982	1986	1987#	1991	1996	2000	2001#	2006	2011	2015	2016#	2020
		2025	2029	2030#	2038	2042	2046	2047#	2054	2058	2062	2063#	2070	2075
		2079	2080#	2092	2097	2101	2102#	2114	2119	2123	2124#	2132	2137	2141
		2142#	2150	2155	2159	2160#	2169	2174	2178	2179#	2188	2193	2197	2198#
		2206	2210	2214	2215#	2228	2233	2237	2238#	2251	2256	2260	2261#	2268
		2273	2277	2278#	2290	2295	2299	2300#	2316	2321	2325	2326#	2345	2351
		2355	2356#	2365	2370	2374	2375#	2383	2388	2392	2393#	2399	2404	2408
		2409#	2417	2422	2426	2427#	2436	2441	2445	2446#	2457	2462	2466	2467#
		2478	2482	2486	2487#	2498	2503	2507	2508#	2519	2524	2528	2529#	2541
		2546	2550	2551#	2564	2570	2571	2575#	2587	2593	2597	2598#	2610	2616
		2620	2621#	2637	2641	2642#	2646	2653	2657	2658#	2667	2672	2676	2677#
		2685	2690	2694	2695#	2708	2713	2717	2718#	2725	2730	2734	2735#	2745
		2749	2750#	2758	2763	2767	2774#	2775	2780	2784	2785#	2800	2805	2809
		2810#	2825	2830	2834	2835#	2841	2849	2853	2854#	2869	2873	2877	2878#
		2891	2836	2900	2901#	2908	2912	2916	2917#	2925	2930	2934	2935#	2951
		2956	2960	2961#	2979	2984	2988	2989#	2997	3002	3006	3007#	3015	3020
		3024	3025#	3033	3038	3042	3043#	3050	3055	3059	3060#	3079	3083	3084#
		3090	3095	3099	3100#	3108	3113	3117	3118#	3127	3132	3136	3137#	3159
		3164	3168	3169#	3195	3201	3205	3206#	3232	3238	3242	3243#	3269	3275
		3279	3280#	3306	3312	3316	3317#	3333	3337	3341	3342#	3358	3362	3366
		3367#	3383	3387	3391	3392#	3408	3413	3417	3418#	3434	3439	3443	3444#
		3460	3465	3469	3470#	3486	3491	3495	3496#	3512	3517	3521	3522#	3535
		3539	3543	3544#	3557	3562	3566	3567#	3598	3602	3603#	3634	3638	3639#
		3648	3652	3656	3657#	3666	3670	3674	3675#	3684	3688	3692	3693#	3706
		3711	3715	3716#	3722	3727	3731	3732#	3750	3755	3759	3760#	3764	3769
		3773	3774#	3778	3783	3787	3788#	3792	3797	3801	3802#	3807	3812	3816
		3817#	3822	3827	3831	3832#	3837	3842	3846	3847#	3851	3856	3860	3861#
		3866	3871	3875	3876#	3881	3886	3890	3891#	3896	3901	3905	3906#	3911
		3916	3920	3921#	3926	3931	3935	3936#	3941	3946	3950	3951#	3955	3960
		3964	3965#	3970	3975	3979	3980#	3985	3990	3994	3995#	4000	4005	4009
		4010#	4015	4020	4024	4025#	4029	4034	4038	4039#	4044	4049	4053	4054#
		4059	4064	4068	4069#	4074	4079	4083	4084#	4100	4105	4109	4110#	4118
		4123	4127	4128#	4135	4140	4144	4145#	4171	4176	4180	4181#	4192	4197
		4201	4202#	4216	4221	4225	4226#	4247	4252	4256	4257#	4268	4274	4278
		4279#	4308	4314	4318	4319#	4335	4339	4340#	4369	4375	4379	4380#	4398
		4402	4403#	4415	4420	4424	4425#	4435	4439	4443	4444#	4463	4467	4468#
		4479	4484	4488	4489#	4512	4516	4517#	4525	4533	4537	4538#	4545	4551
		4555	4556#	4563	4569	4581	4582#	4649	4654	4656#	4659	4663	4668	4670#
		4673	4677	4682	4684#	4693	4697	4702	4704#	4715	4719	4724	4726#	4735
		4739	4744	4746#	4757	4761	4766	4768#	4771	4775	4780	4782#	4786	4790
		4795	4797#	4801	4805	4810	4812#	4816	4820	4825	4827#	4831	4835	4840

4842# 4845 4849 4854 4856# 4860 4864 4869 4871# 4875 4879 4884 4886#
4890 4894 4899 4901# 4916 4920 4925 4927# 4937 4941 4946 4948# 4963
4967 4972 4974# 4980 4984 4989 4991# 5032 5036 5041 5043# 5049 5053
5058 5060# 5075 5079 5084 5086# 5093 5097 5102 5104# 5119 5123 5128
5130# 5145 5149 5154 5156# 5197 5201 5206 5208# 5224 5228 5233 5235#
5250 5254 5259 5261# 5276 5280 5285 5287# 5302 5306 5311 5313# 5329
5333 5338 5340# 5356 5360 5365 5367# 5383 5387 5392 5394# 5410 5414
5419 5421# 5436 5440 5445 5447# 5462 5466 5471 5473# 5488 5492 5497
5499# 5514 5518 5523 5525# 5540 5544 5549 5551# 5566 5570 5575 5577#
5593 5597 5602 5604# 5620 5625 5630 5632# 5648 5653 5658 5660# 5676
5681 5686 5688# 5704 5709 5714 5716# 5732 5737 5742 5744# 5762 5767
5772 5774# 5[illegible] 5793 5798 5800# 5823 5828 5833 5835# 5849 5854 5859
5861# 5[illegible] 5[illegible] 5930 5[illegible]# 5[illegible] 5911 5916 5918# 5936 5941 5946 5948#
5961 5[illegible] 5971 5973# 5[illegible] 5992 5997 5999# 6015 6019 6024 6026# 6041
6045 6050 6052# 6066 6070 6075 6077# 6092 6096 6101 6103# 6118 6122
6127 6129# 6144 6148 6153 6155# 6170 6174 6179 6181# 6196 6200 6205
6207# 6222 6226 6231 6233# 6247 6251 6256 6258# 6273 6277 6282 6284#
6299 6303 6308 6310# 6325 6329 6334 6336# 6351 6355 6360 6362# 6376
6380 6385 6387# 6402 6406 6411 6413# 6428 6432 6437 6439# 6454 6459
6463 6465# 6480 6484 6489 6491# 6506 6510 6515 6517# 6532 6536 6541
6543# 6558 6562 6567 6569# 6584 6588 6593 6595# 6611 6616 6621 6623#
6640 6645 6650 6652# 6672 6677 6682 6684# 6704 6709 6714 6716# 6731
6736 6741 6743# 6758 6763 6768 6770# 6785 6790 6795 6797# 6812 6817
6822 6824# 6839 6844 6849 6851# 6866 6871 6876 6878# 6892 6897 6902
6904# 6919 6924 6929 6931# 6946 6951 6956 6958# 6973 6978 6983 6985#
7000 7005 7010 7012# 7027 7032 7037 7039# 7054 7059 7064 7066# 7081
7086 7091 7093# 7108 7113 7118 7120# 7135 7140 7145 7147# 7162 7167
7172 7174# 7189 7194 7199 7201# 7216 7221 7226 7228# 7243 7248 7253
7255# 7270 7274 7279 7281# 7296 7300 7305 7307# 7322 7326 7331 7333#
7348 7352 7357 7359# 7374 7378 7383 7385# 7400 7405 7410 7412# 7422
7427 7432 7434# 7448 7453 7458 7460# 7469 7474 7479 7481# 7496 7501
7506 7508# 7522 7527 7532 7534# 7548 7553 7558 7560# 7569 7574 7579
7581# 7597 7601 7606 7608# 7624 7628 7633 7635# 7648 7652 7657 7659#
7672 7676 7681 7683# 7697 7701 7706 7708# 7721 7725 7730 7732# 7746
7750 7755 7757# 7765 7769 7774 7776# 7785 7789 7794 7796# 7805 7810
7815 7817# 7826 7831 7836 7838# 7856 7860 7865 7867# 7885 7889 7894
7896# 7914 7918 7923 7925# 7943 7947 7952 7954# 7972 7976 7981 7983#
8001 8005 8010 8012# 8030 8034 8039 8041# 8059 8063 8068 8070# 8080
8085 8090 8092# 8102 8107 8112 8114# 8124 8129 8134 8136# 8146 8151
8156 8158# 8173 8177 8182 8184# 8200 8204 8209 8211# 8227 8231 8236
8238# 8254 8258 8263 8265# 8282 8287 8292 8294# 8311 8316 8321 8323#
8344 8349 8354 8356# 8373 8378 8383 8385# 8401 8405 8410 8412# 8428
8432 8437 8439# 8455 8459 8464 8466# 8482 8486 8491 8493# 8509 8513
8518 8520# 8537 8542 8547 8549# 8566 8571 8576 8578# 8594 8599 8604
8606# 8622 8626 8631 8633# 8649 8653 8658 8660# 8676 8680 8685 8687#
8705 8710 8715 8717# 8736 8741 8746 8748# 8763 8767 8772 8774# 8789
8793 8798 8800# 8815 8819 8824 8826# 8841 8845 8850 8852# 8861 8866
8871 8873# 8882 8887 8892 8894# 8913 8919 8924 8926# 8945 8951 8956
8958# 8977 8983 8988 8990# 9009 9015 9020 9022# 9046 9052 9057 9059#
9083 9089 9094 9096# 9120 9126 9131 9133# 9157 9163 9168 9170# 9194
9200 9205 9207# 9231 9237 9242 9244# 9272 9278 9283 9285# 9309 9315
9320 9322# 9341 9347 9352 9354# 9373 9379 9384 9386# 9405 9411 9416
9418# 9437 9443 9448 9450# 9459 9464 9469 9471# 9480 9485 9490 9492#
9501 9506 9511 9513# 9522 9527 9532 9534# 9543 9549 9554 9556# 9565
9570 9575 9577# 9593 9597 9602 9604# 9621 9625 9630 9632# 9641 9645
9650 9652# 9660 9664 9669 9671# 9684 9688 9693 9695# 9708 9712 9717

CROSS REFERENCE TABLE -- USER SYMBOLS

9719# 9728 9733 9738 9740# 9756 9760 9765 9767# 9782 9787 9792 9794#
9809 9814 9819 9831# 9835 9841 9846 9858 9868 9873 9878# 9880
9 5 9 0 9 2# 9 2 9 7 9912 9914# 9924 9929 9934 9936# 9946 9951
9956 9963# 9968 9973 9983 9990# 9994 9995 10000 10002# 10010 10022 10027
10029# 10049 10053 10058 10060 10078 10080 10085 10087# 10103 10107 10112 10114#
10130 10135 10140 10142# 10148 10162 10167 10169# 10185 10189 10194 10196# 10212
10216 10221 10223# 10232 10242 10247 10249# 10256 10271 10276 10278# 10295 10300
10305 10307# 10324 10333 10334 10336# 10353 10358 10363 10365# 10382 10387 10392
10394# 10411 10415 10420 10422# 10438 10444 10449 10451# 10468 10473 10478 10480#
10496 10501 10506 10508# 10524 10528 10533 10535# 10551 10555 10560 10562# 10579
10583 10588 10590# 10606 10610 10615 10617# 10633 10637 10642 10644# 10660 10663
10668 10670# 10686 10690 10695 10697# 10713 10717 10722 10724# 10740 10744 10749
10751# 10768 10773 10778 10782 10797 10802 10807 10809# 10827 10832 10837 10839#
10857 10862 10867 10869# 10892 10902 10907 10909# 10921 10926 10931 10933# 10951
10956 10961 10963# 10981 10986 10991 10993# 11010 11015 11020 11022# 11030 11034
11039 11041# 11050 11055 11060 11070# 11072 11077 11082 11084# 11094 11099 11104
11106# 11116 11121 11126 11130# 11133 11143 11148 11150# 11160 11165 11170 11172#
11182 11187 11192 11194# 11203 11213 11213 11215# 11225 11230 11235 11237# 11246
11251 11256 11258# 11268 11273 11280 11286 11288# 11291 11295 11300 11302# 11308
11313 11317 11322 11324# 11330 11340 11344 11349 11351# 11357 11362 11366 11371
11373# 11379 11389 11392 11398 11403 11405# 11411 11416 11419 11424 11429 11431#
11437 11441 11451 11455 11460 11462# 11468 11473 11477 11482 11484# 11490 11500
11503 11507 11512 11514# 11520 11525 11528 11533 11538 11540# 11546 11551 11554
11559 11564 11566# 11572 11577 11580 11585 11590 11592# 11598 11601 11606 11609
11615 11620 11622# 11628 11631 11636 11639 11645 11650 11652# 11664 11673 11678
11680# 11704 11709 11711# 11733 11738 11740# 11758 11763 11765# 11777 11787 11792
11794# 11806 11820 11825 11827# 11842 11852 11857 11859# 11871 11885 11890 11892#
11904 11913 11918 11920# 11933 11947 11952 11954# 11958 11963 11968 11969# 11984
11987 11990 11995 11997# 12005 12009 12014 12016# 12024 12028 12033 12035# 12043
12047 12052 12053# 12062 12066 12071 12073# 12099 12107 12112 12114# 12136 12144
12149 12151# 12173 12181 12186 12188# 12200 12208 12220 12229 12241 12246 12248#
12260 12268 12280 12289 12301 12306 12308# 12325 12330 12332# 12338 12343 12348
12350# 12356 12361 12366 12368# 12391 12396 12398# 12423 12428 12430# 12456 12461
12463# 12489 12494 12496# 12546 12551 12553# 12603 12608 12610# 12634 12639 12641#
12665 12670 12672# 12696 12701 12703# 12727 12732 12734# 12758 12763 12765# 12794
12799 12801# 12826 12834 12836# 12870 12879 12881# 12957 12962 12964# 12977 12982
12984# 13017 13022 13024# 13055 13060 13062# 13080 13085 13087# 13105 13110 13112#
13130 13135 13137# 13154 13159 13161# 13188 13193 13198 13200# 13226 13231 13236
13238# 13265 13283 13288 13290# 13334 13343 13348 13350# 13394 13403 13408 13410#
13433 13438 13440# 13465 13470 13472# 13497 13502 13504# 13520 13525 13527# 13543
13548 13550# 13565 13570 13572# 13596 13603 13605# 13619 13633 13638 13640# 13654
13668 13673 13675# 13689 13703 13708 13709# 13729 13734 13735# 13756 13761 13762#
13783 13788 13789# 13813 13818 13819# 13853 13860 13861# 13883 13 3 13890# 13898
13903 13905# 13923 13928 13930# 13939 13944 13946# 13956 13961 13 3# 13976 14010
14011# 14036 14070 14071# 14096 14130 14131# 14156 14190 14191# 14216 14236 14237#
14252 1 .261 14280 14281# 14296 14305 14310 14311# 14331 14335 14340 14341# 14362
14366 14371 14372# 14398 14402 14407 14408# 14434 14438 14443 14444# 14470 14474
14479 14480# 14506 14510 14515 14516# 14537 14541 14546 14547# 14561 14565 14570
14571# 14590 14594 14599 14600# 14619 14623 14628 14629# 14648 14652 14657 14658#
14670 14674 14683 14685# 14713 14720 14725 14727# 14741 14744 14749 14751# 14766
14769 14774 14776# 14790 14793 14798 14799# 14818 14821 14826 14827# 14847 14850
14855 14856# 14879 14895 14903 14905# 14933 14946 14948# 14984 14997 14999# 15030
15041 15043# 15065 15083 15085# 15152 15161 15163# 15213 15224 15226# 15283 15291
15293# 15320 15332 15334# 15397 15408 15410# 15436 15449 15451# 15558 15574 15576#
15645 15649 15651# 15682

$TPB 001052 910# 16192* 16203

```
$TPFLG  001057            914#  16141   16203
$TPS    001050            909#  16190   16203
$TRAP   063166           4412    4629   16348#
$TRAP2  063210          16359#  16370
$TRP    =  [illegible]  16363#  16372#  16373#  16374#  16375#
$TRPAD  [illegible]     16353   16370#
$TSTM   [illegible]       864#
$TSTMM  001002            887#  15928   15942#  15984#  15985   15990   15992   15998   16033   16064
$TYP    *= ****** U       16375
$TYPUS  *= ****** U       16375
$TYPE   062210          16141#  16313   16363   16371
$TYPEC  062422          16171   16178   16185   16190#  16191
$TYPCX  062470          16196   16198   16201#
$TYPOC  062516          16234#  16372
$TYPON  062532          16233   16236#  16374
$TYPOS  062472          16229#  16373
$UNIT   001132            944#
$UNITM  000710            866#
$USWR   001144            951#
$XTSTR  061350          15953#
$$GET4= 000000          15715#
$OFILL  062715          16230#  16234#  16244   16279#
$40CAT= ****** U        15950   16040
.       = 070044          836#    840#    845#    851     852#    854#    856#    872     873#    875#    877#    884#    933
                         2302    2743    2963    3695    4204    4929    5802    6654    7510    8325    9246   10031   10901
                        11713   12412   12623   12654   12685   12716   12747   12767   12782   12814   13907   14858   14930
                        15723   15724#  15745   15772   15997   15998   16064   16114#  16203   16334#  16398#  16582   16583
                        16605#  17125#
.$ASTA= ****** U        16286   16289
.$X     = 000700          851#    856
```

```
COMMEN    8348
ENDCOM    8348
ENDLOP   15304#  16060
ENDPAS   15687#  15706
ENDSCO   15922#  15992
ERROR     7288   4432  4661  4675  4695  4711  4717  4737  4753  4759  4773  4788  4803  4818  4833
              4847  4862  4877  4892  4913  4918  4939  4959  4965  4982  5004  5009  5024  5029  5034
              5051  5072  5077  5096  5116  5121  5142  5147  5169  5174  5190  5194  5199  5221  5226
              5247  5252  5273  5278  5299  5304  5326  5331  5353  5358  5380  5385  5407  5412  5433
              5438  5459  5464  5485  5490  5511  5516  5537  5542  5563  5588  5590  5595  5617  5623
              5645  5651  5673  5679  5701  5707  5729  5735  5754  5759  5765  5785  5791  5815  5830
              5826  5846  5852  5872  5877  5893  5913  5926  5928  5933  5939  5958  5964  5985  5990
              6012  6017  6038  6043  6063  6068  6089  6094  6115  6120  6141  6146  6167  6172  6193
              6198  6219  6224  6244  6249  6270  6275  6296  6301  6322  6327  6348  6353  6374  6378
              6399  6404  6425  6430  6451  6456  6477  6482  6503  6508  6529  6534  6555  6560  6581
              6586  6608  6614  6637  6643  6669  6675  6696  6701  6707  6729  6734  6756  6761  6783
              6788  6810  6815  6837  6842  6864  6869  6890  6895  6917  6922  6944  6949  6971  6976
              6998  7003  7025  7030  7052  7057  7079  7084  7106  7111  7133  7138  7160  7165  7187
              7192  7214  7219  7241  7246  7267  7272  7293  7298  7319  7324  7345  7350  7371  7376
              7397  7403  7425  7445  7451  7472  7493  7499  7525  7545  7551  7572  7594  7595  7621
              7626  7645  7650  7669  7674  7694  7699  7718  7723  7743  7748  7767  7787  7808  7829
              7852  7858  7881  7837  7910  7916  7939  7945  7968  7974  7997  8003  8026  8032  8055
              8061  8083  8105  8127  8149  8170  8175  8197  8202  8224  8229  8251  8256  8279  8285
              8308  8314  8341  8347  8370  8376  8398  8403  8425  8430  8452  8457  8479  8484  8506
              8511  8534  8540  8563  8569  8591  8597  8619  8624  8646  8651  8673  8678  8702  8708
              8733  8739  8760  8765  8786  8791  8812  8817  8838  8843  8864  8885  8910  8917  8942
              8949  8974  8981  9006  9013  9038  9043  9050  9075  9080  9087  9112  9117  9124  9149
              9154  9161  9186  9191  9198  9223  9228  9235  9264  9269  9276  9301  9306  9313  9338
              9345  9370  9377  9402  9409  9434  9441  9462  9483  9504  9525  9546  9568  9590  9595
              9618  9623  9643  9662  9681  9686  9705  9710  9731  9753  9758  9779  9785  9806  9812
              9833  9839  9861  9883  9905  9927  9949  9971  9993  10015 10020 10046 10051 10073 10078
             10100 10105 10127 10133 10155 10160 10182 10187 10209 10214 10235 10240 10263 10269 10292
             10298 10321 10327 10350 10356 10379 10385 10408 10414 10436 10442 10465 10471 10493 10499
             10521 10526 10548 10553 10576 10581 10603 10608 10630 10635 10657 10662 10683 10688 10710
             10715 10737 10742 10765 10771 10794 10800 10824 10830 10854 10860 10884 10890 10918 10924
             10948 10954 10978 10984 11007 11013 11032 11053 11075 11097 11119 11141 11163 11185 11206
             11228 11249 11271 11285 11293 11307 11315 11329 11337 11342 11356 11364 11378 11386 11391
             11394 11410 11418 11422 11436 11440 11448 11453 11467 11475 11489 11497 11502 11505 11519
             11527 11531 11545 11553 11556 11571 11579 11582 11597 11600 11608 11613 11627 11630 11638
             11643 11660 11670 11690 11695 11700 11723 11730 11748 11754 11773 11783 11802 11812 11816
             11835 11838 11848 11867 11877 11881 11900 11910 11928 11939 11943 11957 11986 11988 12007
             12026 12045 12064 12083 12091 12096 12103 12126 12133 12140 12163 12170 12177 12198 12206
             12219 12227 12237 12258 12266 12279 12287 12297 12320 12341 12359 12384 12415 12443 12451
             12476 12484 12519 12526 12535 12538 12576 12583 12592 12595 12626 12657 12688 12719 12750
             12786 12818 12860 12864 12919 12926 12933 12939 12944 12972 12997 13005 13012 13036 13044
             13051 13075 13100 13125 13148 13169 13177 13185 13208 13216 13223 13246 13254 13262 13279
             13301 13311 13321 13331 13361 13371 13381 13391 13419 13429 13451 13460 13483 13492 13515
             13538 13561 13583 13590 13618 13628 13653 13663 13688 13698 13721 13747 13774 13802 13831
             13878 13896 13920 13937 13954 13974 14029 14039 14149 14209 14250 14294 14326 14333 14357
             14364 14388 14395 14400 14424 14431 14436 14460 14467 14472 14496 14503 14508 14532 14539
             14563 14585 14592 14614 14621 14643 14650 14672 14718 14738 14742 14763 14767 14787 14791
             14813 14819 14842 14848 14874 14880 14914 14918 14931 14975 15022 15061 15104 15113 15134
             15171 15181 15186 15205 15248 15277 15304 15315 15347 15354 15362 15376 15384 15392 15414
             15425 15431 15468 15484 15493 15504 15515 15530 15538 15552 15600 15611 15634 15664 15679
             15940
ESCAPE    8348#
```

```
GETPRI    834#
GETSWR    834#
MSG      4569#  4571  12026# 12028  12070# 12072  13596# 13598  13853# 13855  13976# 13978  14036# 14038  14096#
        14098  14156# 14158  14216# 14218  14261# 14263  14674# 14676  15151# 15154  15212# 15215  15282# 15285
        15319# 15322  15396# 15399  15435# 15438  15557# 15560
MSGJ     14895# 14897
MSGM1    14932# 14935
MSGM10   15029# 15032
MSGM11   15064# 15067
MSGM3    14984# 14986
MULT      834#
NEWTST    834#  1943  1966  1989  1992  1996  2011  2025  2042  2058  2075  2097  2119  2137  2155
          2174  2193  2210  2227  2273  2295  2321  2351  2370  2388  2404  2422  2441  2462
          2482  2503  2524  2543  2616  2637  2653  2672  2690  2713  2730  2745  2763
          2780  2805  2826  2873  2896  2912  2930  2946  2984  3002  3020  3038  3055  3079
          3095  3113  3134  3201  3238  3275  3312  3337  3362  3387  3413  3439  3465  3491
          3517  3539  3560  3634  3652  3670  3688  3711  3727  3755  3769  3763  3797  3812
          3827  3842  3861  3871  3888  3901  3916  3931  3946  3960  3975  3990  4005  4020  4034
          4049  4064  4078  4106  4123  4140  4176  4197  4221  4252  4274  4314  4335  4375  4398
          4420  4439  4463  4484  4512  4533  4551  4569  4649  4663  4677  4697  4719  4739  4761
          4775  4790  4805  4820  4835  4849  4864  4879  4894  4920  4941  4967  4984  5036  5053
          5079  5097  5123  5146  5201  5228  5254  5280  5306  5333  5360  5387  5414  5440  5466
          5492  5518  5544  5570  5597  5625  5653  5681  5709  5737  5767  5793  5828  5854  5885
          5911  5941  5966  5992  6019  6045  6070  6096  6122  6148  6174  6200  6226  6251  6277
          6303  6329  6355  6380  6406  6432  6458  6484  6510  6536  6562  6588  6616  6645  6677
          6709  6736  6763  6790  6817  6844  6871  6897  6924  6951  6978  7005  7032  7059  7086
          7113  7140  7167  7194  7221  7248  7274  7300  7326  7352  7378  7405  7427  7453  7474
          7501  7527  7553  7574  7601  7628  7652  7676  7701  7725  7750  7769  7789  7810  7831
          7860  7889  7918  7947  7976  8005  8034  8063  8085  8107  8129  8151  8177  8204  8231
          8258  8287  8316  8349  8378  8405  8432  8459  8486  8513  8542  8571  8599  8626  8653
          8680  8710  8741  8767  8793  8819  8845  8866  8887  8919  8951  8983  9015  9052  9089
          9126  9163  9200  9237  9278  9315  9347  9379  9411  9443  9464  9485  9506  9527  9549
          9570  9597  9625  9645  9664  9688  9712  9733  9760  9787  9814  9841  9863  9885  9907
          9949  9951  9973  9995  10022 10053 10080 10107 10135 10162 10189 10216 10242 10271 10300
         10329 10358 10387 10415 10444 10473 10501 10528 10555 10583 10610 10637 10663 10690 10717
         10744 10773 10802 10832 10862 10892 10926 10956 10986 11015 11034 11055 11077 11099 11121
         11143 11165 11187 11208 11230 11251 11273 11295 11317 11344 11366 11398 11424 11455 11477
         11507 11533 11559 11585 11615 11645 11673 11704 11733 11758 11787 11820 11852 11885 11913
         11947 11963 11990 12009 12028 12047 12066 12107 12144 12181 12241 12301 12325 12343 12361
         12391 12423 12456 12489 12546 12603 12634 12665 12696 12727 12758 12794 12826 12870 12957
         12977 13017 13055 13080 13105 13130 13154 13193 13231 13283 13343 13403 13433 13465 13497
         13520 13543 13565 13596 13633 13668 13703 13729 13756 13783 13813 13853 13883 13898 13923
         13939 13956 13976 14036 14096 14156 14216 14261 14305 14335 14366 14402 14438 14474 14510
         14541 14565 14594 14623 14652 14674 14720 14744 14769 14793 14821 14850 14895 14933 14984
         15030 15065 15152 15213 15283 15320 15397 15436 15558 15645
NOINST    1270#  11966 12050 13706 13732 13759 13786 13816 14008 14068 14128 14188 14234 14278 14308
         14338 14369 14405 14441 14477 14513 14544 14568 14597 14626 14655 14796 14824 14853
NOSCOP    1270#  1946  1959  1973  1985  1999  2014  2028  2045  2061  2078  2100  2122  2140  2158
          2177  2196  2213  2236  2259  2276  2298  2324  2354  2373  2391  2407  2425  2444  2465
          2485  2506  2527  2549  2573  2596  2619  2640  2656  2675  2693  2716  2733  2748  2766
          2783  2808  2833  2852  2876  2899  2915  2933  2959  2987  3005  3023  3041  3058  3082
          3098  3116  3135  3167  3204  3241  3278  3315  3340  3365  3390  3416  3442  3468  3494
          3520  3542  3565  3601  3637  3655  3673  3691  3714  3730  3758  3772  3786  3800  3815
          3830  3845  3859  3874  3889  3904  3919  3934  3949  3963  3978  3993  4008  4023  4037
          4052  4067  4082  4108  4126  4143  4179  4200  4224  4255  4277  4317  4338  4378  4401
          4423  4442  4466  4487  4515  4536  4554  4580  13859
```

```
POP       834#  15757  15758  16328  16329
PRTERR  16003#  16019
PRTNEW   1270#  14902  14945  14996  15040  15082  15160  15223  15290  15331  15407  15448  15573  15648
PRESCO  15921#  15937
PUSH      834#  15735  15741  16289  16291  16312
REPORT    834#
SCOPE    7298#  4326   4653   4667   4681   4701   4723   4743   4765   4779   4794   4809   4824   4839   4853
          4868   4883   4898   4924   4945   4971   4988   5040   5057   5083   5101   5127   5153   5205   5232
          5258   5284   5310   5337   5[illegible]4   5391   5418   5444   5470   5496   5522   5[illegible]8   5574   5601   5629
          5657   5685   5713   5741   5771   5797   5832   5858   5889   5915   5945   59[illegible]0   5996   6023   6049
          6074   6100   6126   6152   6178   6204   6230   6255   6281   6307   6333   6359   6384   6410   6436
          6462   6488   6514   6540   6[illegible]6   6[illegible]2   6[illegible]0   6649   6[illegible]1   6713   6740   6[illegible]7   6794   6821   6848
          6875   6901   6928   6[illegible]5   6[illegible]3   7009   7036   7063   7[illegible]0   7117   7144   [illegible]1   7198   7225   7252
          7278   7304   7330   73[illegible]6   7[illegible]2   7409   7431   7457   74[illegible]8   75[illegible]5   75[illegible]1   [illegible]   75[illegible]8   7605   7632
          7656   7680   7705   7729   7754   7773   7793   7814   7835   7[illegible]4   7[illegible]3   7[illegible]   79[illegible]1   7960   8009
          8038   8067   8089   8111   8133   8155   8181   8208   82[illegible]5   8[illegible]2   8[illegible]1   8[illegible]   8353   8382   8409
          8436   8463   8490   8517   8546   8575   8603   8630   8657   8684   8714   8[illegible]5   8771   8797   8823
          8849   8870   8891   8923   8[illegible]5   8987   9019   9[illegible]6   9[illegible]3   9130   9167   9[illegible]4   9241   9282   9319
          9351   9383   9415   9447   9468   9489   9510   9531   9[illegible]3   9574   9601   9[illegible]9   9649   9668   9692
          9716   973[illegible]   9764   9791   9818   9845   9867   9899   9911   9933   9955   9977   9999   10[illegible]26   10057
         10084  10111  10139  10166  10193  10220  10246  10275  10304  10333  10362  10391  10419  10448  10477
         10505  10532  10559  10587  10614  10641  10667  10694  10721  10748  10777  10806  10836  10866  10896
         10930  10960  10990  11019  11038  11059  11081  11103  11125  11147  11169  11191  11212  11234  11255
         11277  11299  11321  11348  11370  11402  11423  11459  11481  11511  11537  11563  11589  11619  11649
         11677  11708  11737  11762  11791  11824  11856  11889  11917  11951  11967  11994  12013  12032  12051
         12070  12111  12148  12185  12245  12305  12329  12347  12365  12395  12427  12460  12493  12550  12607
         12638  12669  12700  12731  12762  12798  12833  128[illegible]8  12961  12981  13021  13059  13084  13109  13134
         13158  13197  13235  13287  13347  13407  13437  13469  13501  13524  13547  13[illegible]9  13602  13637  13672
         13707  13733  13760  13787  13817  13844  13887  13902  13927  13943  13960  14009  14069  14129  14189
         14235  14279  14309  14339  14370  14406  14442  14478  14514  14545  14569  14598  14627  14656  14682
         14724  14748  14773  14797  14825  14854  14904  14947  14998  15042  15084  15162  15225  15292  15333
         15409  15450  15575  15650  156[illegible]6
SETPRI    834#
SETTRA  1636[illegible]#  16372  16373  16374
SETUP     834#
SKIP      834#  1950   1965   1977   1991   2006   2019   2038   2054   2070   2092   2114   2132   2150   2169
          2188   2206   2228   2251   2268   2290   2316   2345   2365   2383   2399   2417   2436   2457   2478
          2498   2519   2541   2564   2587   2610   2648   2667   2685   2708   2725   2758   2775   2800   2825
          2844   2869   2891   2908   2925   2951   2979   2997   3015   3033   3050   3090   3108   3127   3159
          3195   3232   3269   3306   3333   3358   3383   3408   3434   3460   3486   3512   3535   3557   3648
          3666   3684   3706   3722   3749   3764   3777   3791   3807   3822   3836   3851   3865   3880   3896
          3911   3926   3941   3954   3970   3985   4000   4014   4028   4044   4059   4074   4100   4118   4135
          4171   4192   4216   4247   4268   4308   4369   4415   4435   4479   4525   4545   4563   4659   4672
          4693   4715   4735   4757   4771   4785   4800   4815   4831   4845   4859   4874   4889   4916   4937
          49[illegible]   4980   5032   5049   5075   5093   5119   5145   5197   5224   5250   5276   5302   5329   5356
          5[illegible]3   5410   5436   5462   5488   5514   5540   55[illegible]   5593   5620   5648   5676   5704   5732   5762
          5788   5823   5849   5830   5906   5936   5961   5988   6015   6041   6066   6092   6118   6144   6170
          6196   6222   6247   6273   6299   6325   6351   6376   6402   6428   6454   6480   6506   6532   6558
          6584   6611   6640   6672   6704   6731   6758   6785   6812   6839   6866   6892   6919   6946   6973
          7000   7027   7054   7081   7108   7135   7162   7189   7216   7243   7270   7296   7322   7348   7374
          7400   7422   7448   7469   7496   7522   7548   7569   7597   7624   7648   7672   7697   7721   7746
          7765   7785   7805   7826   7856   7885   7914   7943   7972   8001   8030   8059   8080   8102   8124
          8146   8173   8200   8227   8254   8282   8311   8344   8373   8401   8428   8455   8482   8509   8537
          8566   8594   8622   8649   8676   8705   8736   8763   8789   8815   8841   8861   8882   8913   8945
          8977   9009   9046   9083   9120   9157   9194   9231   9272   9309   9341   9373   9405   9437   9459
          9480   9501   9522   9543   9565   9593   9621   9641   9660   9684   9708   9728   9756   9782   9809
```

	9836	9858	9880	9902	9924	9946	9968	9990	10018	10049	10076	10103	10130	10158	10185
	10212	10238	10266	10295	10324	10353	10382	10411	10439	10468	10496	10524	10551	10579	10606
	10633	10660	10686	10713	10740	10768	10797	10827	10857	10887	10921	10951	10981	11010	11030
	11050	11072	11094	11116	11138	11160	11182	11203	11225	11246	11268	11286	11291	11308	11313
	11330	11340	11357	11362	11379	11389	11392	11411	11416	11419	11437	11441	11451	11468	11473
	11490	11500	11503	11520	11525	11528	11546	11551	11554	11572	11577	11580	11598	11601	11606
	11609	11628	11631	11636	11639	11664	11777	11806	11842	11871	11904	11933	11958	11984	11987
	12005	12024	12043	12062	12099	12136	12173	12200	12208	12220	12229	12250	12268	12280	12289
	12338	12356	13138	13226	13265	13334	13394	13619	13654	13689	14252	14236	14331	14362	14398
	14434	14470	14506	14537	14561	14590	14619	14648	14670	14713	14741	14766	14790	14818	14847
	14879														
SLASH	8348														
SPACE	8348														
STARS	8348	848	850	857	870	880	933	936	1943	1945	1956	1958	1970	1972	1982
	1984	1996	1998	2011	2013	2025	2027	2042	2044	2058	2060	2075	2077	2097	2099
	2119	2121	2137	2139	2155	2157	2174	2176	2193	2195	2210	2212	2233	2235	2256
	2258	2273	2275	2295	2297	2321	2323	2351	2353	2370	2372	2388	2390	2404	2406
	2422	2424	2441	2443	2462	2464	2482	2484	2503	2505	2524	2526	2546	2548	2570
	2572	2593	2595	2616	2618	2637	2639	2653	2655	2672	2674	2690	2692	2713	2715
	2730	2732	2745	2747	2763	2765	2780	2782	2805	2807	2830	2832	2849	2851	2873
	2875	2896	2898	2912	2914	2930	2932	2956	2958	2984	2986	3002	3004	3020	3022
	3038	3040	3055	3057	3079	3081	3095	3097	3113	3115	3132	3134	3164	3166	3201
	3203	3238	3240	3275	3277	3312	3314	3337	3339	3362	3364	3387	3389	3413	3415
	3439	3441	3465	3467	3491	3493	3517	3519	3539	3541	3562	3564	3538	3600	3634
	3636	3652	3654	3670	3672	3688	3690	3711	3713	3727	3729	3755	3757	3769	3771
	3783	3785	3797	3799	3812	3814	3827	3829	3842	3844	3856	3858	3871	3873	3886
	3888	3901	3903	3916	3918	3931	3933	3946	3948	3960	3962	3975	3977	3990	3992
	4005	4007	4020	4022	4034	4036	4049	4051	4064	4066	4079	4081	4105	4107	4123
	4125	4140	4142	4176	4178	4197	4199	4221	4223	4252	4254	4274	4276	4314	4316
	4335	4337	4375	4377	4398	4400	4420	4422	4439	4441	4463	4465	4484	4486	4512
	4514	4533	4535	4551	4553	4569	4579	4649	4651	4663	4665	4677	4679	4697	4699
	4719	4721	4739	4741	4761	4763	4775	4777	4790	4792	4805	4807	4820	4822	4835
	4837	4849	4851	4864	4866	4879	4881	4894	4896	4920	4922	4941	4943	4967	4969
	4984	4986	5036	5038	5053	5055	5079	5081	5097	5099	5123	5125	5149	5151	5201
	5203	5228	5230	5254	5256	5280	5282	5306	5308	5333	5335	5360	5362	5387	5389
	5414	5416	5440	5442	5466	5468	5492	5494	5518	5520	5544	5546	5570	5572	5597
	5599	5625	5627	5653	5655	5681	5683	5709	5711	5737	5739	5767	5769	5793	5795
	5828	5830	5854	5856	5885	5887	5911	5913	5941	5943	5966	5968	5992	5994	6019
	6021	6045	6047	6070	6072	6096	6098	6122	6124	6148	6150	6174	6176	6200	6202
	6226	6228	6251	6253	6277	6279	6303	6305	6329	6331	6355	6357	6380	6382	6406
	6408	6432	6434	6460	6484	6486	6510	6512	6536	6538	6562	6564	6588	6590	
	6616	6618	6677	6679	6709	6711	6736	6738	6763	6765	6790	6792	6817		
	6819	6844	6897	6899	6924	6926	6951	6953	6978	6980	7005	7007			
	7032	7034	7088	7113	7115	7140	7142	7167	7169	7194	7196	7221			
	7223	7248	7300	7302	7326	7328	7352	7354	7378	7380	7405	7407			
	7427	7429	7474	7476	7501	7503	7527	7529	7553	7555	7574	7576	7601		
	7603	7628	7654	7676	7678	7701	7703	7725	7727	7750	7752	7769	7771		
	7789	7791	7810	7831	7833	7860	7862	7889	7891	7918	7920	7947	7949	7976	
	7978	8005	8007	8036	8063	8065	8085	8087	8107	8109	8129	8131	8151	8153	
	8177	8179	8204	8231	8233	8258	8260	8287	8289	8316	8318	8349	8351	8378	
	8380	8405	8407	8434	8459	8461	8486	8488	8513	8515	8542	8544	8571	8573	
	8599	8601	8626	8628	8653	8655	8680	8682	8710	8712	8741	8743	8767	8769	8793
	8795	8819	8821	8845	8847	8866	8868	8887	8889	8919	8921	8951	8953	8983	8985
	9015	9017	9052	9054	9089	9091	9126	9128	9163	9165	9200	9202	9237	9239	9278
	9280	9315	9317	9347	9349	9379	9381	9411	9413	9443	9445	9464	9466	9485	9487
	9506	9508	9527	9529	9549	9551	9570	9572	9597	9599	9625	9627	9645	9647	9664

	9666	9688	9690	9712	9714	9733	9735	9760	9762	9787	9789	9814	9816	9841	9843
	9863	9865	9885	9887	9907	9909	9929	9931	9951	9953	9973	9975	9995	9997	10022
	10024	10053	10055	10080	10082	10107	10109	10135	10137	10162	10164	10189	10191	10216	10218
	10242	10244	10271	10273	10300	10302	10329	10331	10358	10360	10387	10389	10415	10417	10444
	10446	10473	10475	10501	10503	10528	10530	10555	10557	10583	10585	10610	10612	10637	10639
	10663	10665	10690	10692	10717	10719	10744	10746	10773	10775	10802	10804	10832	10834	10862
	10864	10892	10894	10926	10928	10956	10958	10986	10988	11015	11017	11034	11036	11055	11057
	11077	11079	11099	11101	11121	11123	11143	11145	11165	11167	11187	11189	11208	11210	11230
	11232	11251	11253	11273	11275	11295	11297	11317	11319	11344	11346	11366	11368	11398	11400
	11424	11426	11455	11457	11477	11479	11507	11509	11533	11535	11559	11561	11585	11587	11615
	11617	11645	11647	11673	11675	11704	11706	11733	11735	11758	11760	11787	11789	11820	11822
	11852	11854	11885	11887	11913	11915	11947	11949	11963	11965	11990	11992	12009	12011	12028
	12030	12047	12049	12066	12068	12107	12109	12144	12146	12181	12183	12241	12243	12301	12303
	12325	12327	12343	12345	12361	12363	12391	12393	12423	12425	12456	12458	12489	12491	12546
	12548	12603	12605	12634	12636	12665	12667	12696	12698	12727	12729	12758	12760	12794	12796
	12826	12831	12870	12876	12957	12959	12977	12979	13017	13019	13055	13057	13080	13082	13105
	13107	13130	13132	13154	13156	13193	13195	13231	13233	13283	13285	13343	13345	13403	13405
	13433	13435	13465	13467	13497	13499	13520	13522	13543	13545	13565	13567	13596	13600	13633
	13635	13668	13670	13703	13705	13729	13731	13756	13758	13783	13785	13813	13815	13853	13858
	13883	13885	13898	13900	13923	13925	13939	13941	13956	13958	13976	14007	14036	14067	14096
	14127	14156	14187	14216	14233	14261	14277	14305	14307	14335	14337	14366	14368	14402	14404
	14438	14440	14474	14476	14510	14512	14541	14543	14565	14567	14594	14596	14623	14625	14652
	14654	14674	14680	14720	14722	14744	14746	14769	14771	14793	14795	14821	14823	14850	14852
	14895	14901	14933	14944	14984	14995	15030	15039	15065	15081	15152	15159	15213	15222	15283
	15289	15320	15330	15397	15406	15436	15447	15558	15572	15645	15647	15690	15731	15747	15791
	15925	16007	16070	16126	16206	16284	16342								
SWRSU	8348														
TESTNO	12708	1947	1960	1974	1986	2000	2015	2029	2046	2062	2079	2101	2123	2141	2159
	2178	2197	2214	2237	2260	2277	2299	2325	2355	2374	2392	2408	2426	2445	2466
	2486	2507	2528	2550	2574	2597	2620	2641	2657	2676	2694	2717	2734	2749	2767
	2784	2809	2834	2853	2877	2900	2916	2934	2960	2988	3006	3024	3042	3059	3083
	3099	3117	3136	3168	3205	3242	3279	3316	3341	3366	3391	3417	3443	3469	3495
	3521	3543	3566	3602	3638	3656	3674	3692	3715	3731	3759	3773	3787	3801	3816
	3831	3846	3860	3875	3890	3905	3920	3935	3950	3964	3979	3994	4009	4024	4038
	4053	4068	4083	4109	4127	4144	4180	4201	4225	4256	4278	4318	4339	4379	4402
	4424	4443	4467	4488	4516	4537	4555	4581	4654	4668	4682	4702	4724	4744	4766
	4780	4795	4810	4825	4840	4854	4869	4884	4899	4925	4946	4972	4989	5041	5058
	5084	5102	5128	5154	5206	5233	5259	5285	5311	5338	5365	5392	5419	5445	5471
	5497	5523	5549	5575	5602	5630	5658	5686	5714	5742	5772	5798	5833	5859	5890
	5916	5946	5971	5997	6024	6050	6075	6101	6127	6153	6179	6205	6231	6256	6282
	6308	6334	6360	6385	6411	6437	6463	6489	6515	6541	6567	6593	6621	6650	6682
	6714	6741	6768	6795	6822	6849	6876	6902	6929	6956	6983	7010	7037	7064	7091
	7118	7145	7172	7199	7226	7253	7279	7305	7331	7357	7383	7410	7432	7458	7479
	7506	7532	7558	7579	7606	7633	7657	7681	7706	7730	7755	7774	7794	7815	7836
	7865	7894	7923	7952	7981	8010	8039	8068	8090	8112	8134	8156	8182	8209	8236
	8263	8292	8321	8354	8383	8410	8437	8464	8491	8518	8547	8576	8604	8631	8658
	8685	8715	8746	8772	8798	8824	8850	8871	8892	8924	8956	8988	9020	9057	9094
	9131	9168	9205	9242	9283	9320	9352	9384	9416	9448	9469	9490	9511	9532	9554
	9575	9602	9630	9650	9669	9693	9717	9738	9765	9792	9819	9846	9868	9890	9912
	9934	9956	9978	10000	10027	10058	10085	10112	10140	10167	10194	10221	10247	10276	10305
	10334	10363	10392	10420	10449	10478	10506	10533	10560	10588	10615	10642	10668	10695	10722
	10749	10778	10807	10837	10867	10897	10931	10961	10991	11020	11039	11060	11082	11104	11126
	11148	11170	11192	11213	11235	11256	11278	11300	11322	11349	11371	11403	11429	11460	11482
	11512	11538	11564	11590	11620	11650	11678	11709	11738	11763	11792	11825	11857	11890	11918
	11952	11968	11995	12014	12033	12052	12071	12112	12149	12186	12246	12306	12330	12348	12366
	12396	12428	12461	12494	12551	12608	12639	12670	12701	12732	12763	12799	12834	12879	12962

```
            12982  13022  13060  13085  13110  13135  13159  13198  13236  13288  13348  13408  13438  13470  13502
            13525  13548  13570  13603  13638  13673  13708  13734  13761  13788  13818  13860  13888  13903  13928
            13944  13961  14010  14070  14130  14190  14236  14280  14310  14340  14371  14407  14443  14479  14515
            14546  14570  14599  14628  14657  14683  14725  14749  14774  14798  14826  14855  14903  14946  14997
            15041  15083  15161  15224  15291  15332  15408  15449  15574  15649
TRMTRP      16363#
TYPBIN        834#
TYPDEC        834#
TYPNAM        834#
TYPNUM        834#
TYPOCS        834#
TYPOCT        834#  16082  16106
TYPTXT        834#
UPCODE      15729#  15754
YESCOP       1270#   4652   4666   4680   4700   4722   4742   4764   4778   4793   4808   4823   4838   4852   4867
             4882   4897   4923   4944   4970   4987   5039   5056   5082   5100   5126   5152   5204   5231   5257
             5283   5309   5336   5363   5390   5417   5443   5469   5495   5521   5547   5573   5600   5628   5656
             5684   5712   5740   5770   5796   5831   5857   5 8    5914   5944   5969   5995   6022   6048   6073
             6099   6125   6151   6177   6203   6229   6254   6280   6306   6332   6358   6383   6409   6435   6461
             6487   6513   6539   6565   6591   6619   6648   6680   6712   6739   6766   6793   6820   6847   6874
             6900   6927   6954   6981   7008   7035   7062   7089   7116   7143   7170   7197   7224   7251   7277
             7303   7329   7355   7381   7408   7430   7456   7477   7504   7530   7556   7577   7604   7631   7655
             7679   7704   7728   7753   7772   7792   7813   7834   7863   7892   7921   7950   7979   8008   8037
             8066   8088   8110   8132   8154   8180   8207   8234   8261   8290   8319   8352   8381   8408   8435
             8462   8489   8516   8545   8574   8602   8629   8656   8683   8713   8744   8770   8796   8822   8848
             8869   8890   8922   8954   8966   9018   9055   9092   9129   9166   9203   9240   9281   9318   9350
             9382   9414   9446   9467   9488   9509   9530   9552   9573   9600   9628   9648   9667   9691   9715
             9736   9763   9790   9817   9844   9866   9888   9910   9932   9954   9976   9998  10025  10056  10083
            10110  10138  10165  10192  10219  10245  10274  10303  10332  10361  10390  10418  10447  10476  10504
            10531  10558  10586  10613  10640  10666  10693  10720  10747  10776  10805  10835  10865  10895  10929
            10959  10989  11018  11037  11058  11080  11102  11124  11146  11168  11190  11211  11233  11254  11276
            11298  11320  11347  11369  11401  11427  11458  11480  11510  11536  11562  11588  11618  11648  11676
            11707  11736  11761  11790  11823  11855  11888  11916  11950  11993  12012  12031  12069  12110  12147
            12184  12244  12304  12328  12346  12364  12394  12426  12459  12492  12549  12606  12637  12668  12699
            12730  12761  12797  12832  12877  12960  12980  13020  13058  13083  13108  13133  13157  13196  13234
            13286  13346  13406  13436  13468  13500  13523  13546  13568  13601  13636  13671  13886  13901  13926
            13942  13959  14681  14723  14747  14772
SSCMRE        878#   917   918   919   920   921   922
SSCMTM        878#   923   924   925   926   927
SSESCA        834#
SSNEWT        834#  1943  1956  1970  1982  1996  2011  2025  2042  2058  2075  2097  2119  2137  2155
             2174  2193  2210  2233  2256  2273  2295  2321  2351  2370  2388  2404  2422  2441  2462
             2482  2503  2524  2546  2570  2593  2616  2637  2653  2672  2690  2713  2730  2745  2763
             2780  2805  2830  2849  2873  2896  2912  2930  2956  2984  3002  3020  3038  3055  3079
             3095  3113  3132  3164  3201  3238  3275  3312  3337  3362  3387  3413  3439  3465  3491
             3517  3539  3562  3598  3634  3652  3670  3688  3711  3727  3755  3769  3783  3797  3812
             3827  3842  3856  3871  3886  3901  3916  3931  3946  3960  3975  3990  4005  4020  4034
             4049  4064  4079  4105  4123  4140  4176  4197  4221  4252  4274  4314  4335  4375  4398
             4420  4439  4463  4484  4512  4533  4551  4569  4649  4663  4677  4697  4719  4739  4761
             4775  4790  4805  4820  4835  4849  4864  4879  4894  4920  4941  4967  4984  5036  5053
             5079  5097  5123  5149  5201  5228  5254  5280  5306  5333  5360  5387  5414  5440  5466
             5492  5518  5544  5570  5597  5625  5653  5681  5709  5737  5767  5793  5828  5854  5885
             5911  5941  5966  5992  6019  6045  6070  6096  6122  6148  6174  6200  6226  6251  6277
             6303  6329  6355  6380  6406  6432  6458  6484  6510  6536  6562  6588  6616  6645  6677
             6709  6736  6763  6790  6817  6844  6871  6897  6924  6951  6978  7005  7032  7059  7086
             7113  7140  7167  7194  7221  7248  7274  7300  7326  7352  7378  7405  7427  7453  7474
```

```
                7501    7527    7553    7574    7601    7628    7652    7676    7701    7725    7750    7769    7789    7810    7831
                7850    7879    7918    7947    7976    8005    8034    8063    8085    8107    8129    8151    8177    8204    8231
                8258    8287    8316    8349    8378    8405    8432    8459    8486    8513    8542    8571    8599    8626    8653
                8680    8710    8741    8767    8793    8819    8845    8866    8887    8919    8951    8983    9015    9052    9089
                9126    9163    9200    9237    9278    9315    9347    9379    9411    9443    9464    9485    9506    9527    9549
                9570    9597    9625    9645    9664    9688    9712    9733    9760    9787    9814    9841    9863    9885    9907
                9929    9951    9973    9996   10022   10053   10080   10107   10135   10162   10189   10216   10242   10271   10300
               10329   10358   10387   10415   10444   10473   10501   10528   10555   10583   10610   10637   10663   10690   10717
               10744   10773   10802   10832   10862   10892   10926   10956   11015   11034   11055   11077   11099   11121
               11143   11165   11187   11208   11230   11251   11273   11295   11317   11344   11366   11398   11424   11455   11477
               11507   11533   11559   11585   11615   11645   11673   11704   11733   11758   11787   11820   11852   11885   11913
               11947   11963   11990   12009   12028   12047   12066   12107   12144   12181   12241   12301   12325   12343   12361
               12391   12423   12456   12489   12546   12603   12634   12665   12696   12727   12758   12794   12826   12870   12957
               12977   13017   13055   13080   13105   13130   13154   13193   13231   13283   13343   13403   13433   13465   13497
               13520   13543   13565   13596   13633   13668   13703   13729   13756   13783   13813   13853   13883   13898   13923
               13939   13956   13976   14036   14096   14156   14216   14261   14305   14335   14366   14402   14438   14474   14510
               14541   14565   14594   14623   14652   14674   14720   14744   14769   14793   14821   14850   14895   14933   14984
               15030   15065   15152   15213   15283   15320   15397   15436   15558   15645
SSSET          16363#  16372   16373   16374
SSSKIP          834#   1951    1965     978   1991    2006    2020    2038    2054    2070    2092    2114    2132    2150    2169
                2188    2206    2228    2251    2268    2290    2316    2345    2365    2383    2399    2417    2436    2457    2478
                2498    2519    2541    2564    2587    2610    2648    2667    2685    2708    2725    2758    2775    2800    2825
                2844    2869    2891    2908    2925    2951    2979    2997    3015    3033    3050    3090    3108    3127    3159
                3195    3232    3269    3306    3333    3358    3383    3408    3434    3460    3486    3512    3535    3557    3648
                3666    3684    3706    3722    3750    3764    3778    3792    3807    3822    3837    3851    3866    3881    3896
                3911    3926    3941    3955    3970    3985    4000    4015    4029    4044    4059    4074    4100    4118    4135
                4171    4192    4216    4247    4268    4308    4369    4415    4435    4479    4525    4545    4563    4659    4673
                4693    4715    4735    4757    4771    4786    4801    4816    4831    4845    4860    4875    4890    4916    4937
                4963    4980    5032    5049    5075    5093    5119    5145    5197    5224    5250    5276    5302    5329    5356
                5383    5410    5436    5462    5488    5514    5540    5566    5593    5620    5648    5676    5704    5732    5762
                5788    5823    5849    5880    5906    5936    5961    5988    6015    6041    6066    6092    6118    6144    6170
                6196    6222    6247    6273    6299    6325    6351    6376    6402    6428    6454    6480    6506    6532    6558
                6584    6611    6640    6672    6704    6731    6758    6785    6812    6839    6866    6892    6919    6946    6973
                7000    7027    7054    7081    7108    7135    7162    7189    7216    7243    7270    7296    7322    7348    7374
                7400    7422    7448    7469    7496    7522    7548    7569    7597    7624    7648    7672    7697    7721    7746
                7765    7785    7805    7826    7856    7885    7914    7943    7972    8001    8030    8059    8080    8102    8124
                8146    8173    8200    8227    8254    8282    8311    8344    8373    8401    8428    8455    8482    8509    8537
                8566    8594    8622    8649    8676    8705    8736    8763    8789    8815    8841    8861    8882    8913    8945
                8977    9009    9046    9083    9120    9157    9194    9231    9272    9309    9341    9373    9405    9437    9459
                9480    9501    9522    9543    9565    9593    9621    9641    9660    9684    9708    9728    9756    9782    9809
                9836    9858    9880    9902    9924    9946    9968    9990   10018   10049   10076   10103   10130   10158   10185
               10212   10238   10266   10295   10324   10353   10382   10411   10439   10468   10496   10524   10551   10579   10606
               10633   10660   10686   10713   10740   10768   10797   10827   10857   10887   10921   10951   10981   11010   11030
               11050   11072   11094   11116   11138   11160   11182   11203   11225   11246   11268   11286   11291   11308   11313
               11330   11340   11357   11362   11379   11389   11392   11411   11416   11419   11437   11441   11451   11468   11473
               11490   11500   11503   11520   11525   11528   11546   11551   11554   11572   11577   11580   11598   11601   11606
               11609   11628   11631   11636   11639   11664   11777   11806   11842   11871   11904   11933   11958   11984   11987
               12005   12024   12043   12062   12099   12136   12173   12200   12208   12220   12229   12260   12268   12280   12289
               12338   12356   13188   13226   13265   13334   13394   13619   13654   13689   14252   14296   14331   14362   14398
               14434   14470   14506   14537   14561   14590   14619   14648   14670   14713   14741   14766   14790   14818   14847
               14879
.EQUAT          702#    724
.HEADE          702#    703
.SWRHI          702#    712
.SWRLO          723#
.$ACTI          702#    868
```

```
.SAPTB    702#    934#
.SAPTH    702#    846
.SAPTY    702#  16282
.SCATC    702#    834
.SCMTA    702#    878
.SEOP     702#  15688
.SERRO    702#  16005
.SERRT    702#  16068
.SPOKE    702#  15729
.SSCOP    702#  15923
.STRAP    702#  16340
.STYPO    702#
.STYPE    702#  16124
.STYPO    702#  16204


. ABS.  070044     000


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 DSKZ:DQKDAB,DSKZ:DQKDAB.SEQ/SOL/CRF/DS:ERFZ/NL:TOC=DSKZ:DQKDAB.P11
 RUN-TIME: 61 66 5 SECONDS
 RUN-TIME RATIO: 316/133=2.3
 CORE USED:  28K  (56 PAGES)


EOF1DQKDABSEQ              00010000      770624              PDP10 411
```