KD11-K

BASIC LOGIC TEST
CQKDAC0

AH-8092C-MC
COPYRIGHT 77-78
FICHE 1 OF 2

JAN 1979
digital
MADE IN USA

KD11-K

BASIC LOGIC TEST

CQKDAC0

AH-8092C-MC

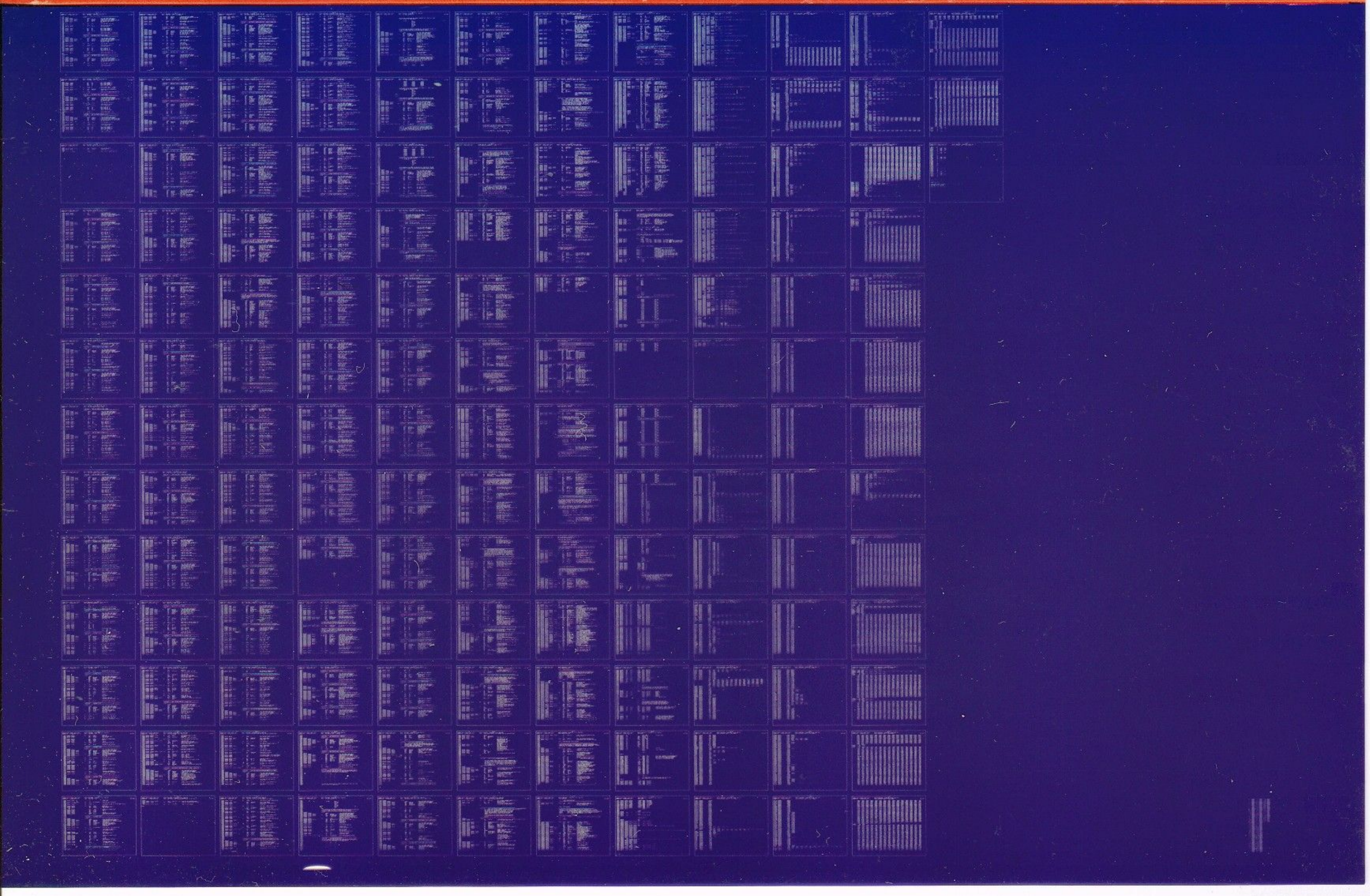
COPYRIGHT 77-78

FICHE 2 OF 2

JAN 1979

digital

MADE IN USA



```
.SBTTL  DOCUMENT LISTING
.TITLE  CQKDAC0, KD11-K BLT
.REM %
```

PRODUCT CODE: AC-8090C-MC

PRODUCT NAME: CQKDAC0 KD11-K BLT

PRODUCT DATE: 15 NOVEMBER 1978

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: JOHN CARMODY

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

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

COPYRIGHT (C) 1977,1978, BY DIGITAL EQUIPMENT CORPORATION.

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

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

''CQKDA'' 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 REP-
ORTS 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 UN-
LESS 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 EXEC-
UTION.

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

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

1.4 DIAGNOSTIC HIERARCHY REQUIREMENTS

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

1.5 FAILURE ASSUMPTIONS

''CQKDA'' 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 ''CQKDA'' WILL NOT PERMIT SIMULTANEOUS RESIDENCY OF THE UPDATE PROGRAMS. SUFFICIENT FREE CORE IS AVAILABLE FOR THE ''XXDP'' MONITOR SO THAT ''CQKDA'' 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 ''CQKDA'' 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, DEPENDING 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 ERROR 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:

                                    CQKDAC  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 ''CQKDA''
                    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:


                                                DATA IN ****
                                                         *
                                                         *
                                                         V
                    **************             *****************
                    *            *             *               *
                    * CONTROL    *>>>>>>>>>>>>>*    DATA       *
                    *  SECTION   *<<<<<<<<<<<<<*     PATHS     *
                    *            *             *               *
```

```
                                    **************          *****************
                                                                  *
                                                                  *
                                                                  ***>DATA OUT

                THE DATA PATHS CONSIST OF A LOGICALLY INTERCONNECTED GROUP
                OF STATIC DATA FACILITIES (REGISTERS, MULTIPLEXORS, ALU'S
                ETC.) REQUIRED TO TEMPORARILY STORE, MODIFY, AND TRANSFER
                DATA ITEMS (16 BIT WORDS OR 8 BIT BYTES) ACCORDING TO THE
                DESIGN SPECIFICATIONS FOR THE PDP11.

                THE CONTROL SECTION SUPPLIES PREDEFINED SEQUENCES OF CONTROL
                SIGNAL SETS TO ACTIVATE THE REQUIRED DATA FACILITIES WITHIN
                THE DATA PATHS. IN THE KD11-K THESE CONTROL SIGNAL SETS ARE
                STORED IN A READ ONLY MEMORY (ROM) AND GENERATED BY READING
                OUT A UNIQUE SEQUENCE OF ROM WORDS FOR EACH OPERATION TO BE
                PERFORMED.

                THE SEQUENCE GENERATED BY THE CONTROL SECTION IS VARIABLE
                AND DEPENDENT UPON THE INSTRUCTION OR LOGIC OPERATION
                BEING EXECUTED. THERE ARE HUNDREDS OF THESE SEQUENCES POSS-
                IBLE DEPENDENT UPON OF THE PROGRAM CODING.

                ''CQKDA'' IS DESIGNED TO GENERATE ALL POSSIBLE MICROINSTRUCTION
                SEQUENCES AND COMBINATIONS OF DATA AND CONTROL SIGNALS. THE
                INDIVIDUAL TESTS ARE LOGICALLY SEQUENCED AND STRUCTURED TO
                DETECT AND ISOLATE PARTICULAR MICROPROGRAM SEQUENCES THAT ARE
                FAULTY.

        5.2     CONDITION CODE SCOPE SYNC FEATURE
                ---------------------------------

                FROM THE BIT SECTION TO THE MED TESTS IN THE CIT SECTION, ALL
                TEST INSTRUCTIONS ARE PRECEEDED BY A CONDITION CODE OPERATE
                INSTRUCTION.  THE UBREAK REGISTER IS PROGRAM LOADED TO GENERATE
                A SYNC PULSE NEAR THE END OF THIS INSTRUCTION.  DURING THE MED
                TESTS, THE PULSE IS GENERATED NEAR THE BEGINNING OF THE MED
                EXECUTION.  THIS PULSE IS GENERATED ON BACKPLANE PIN BO3M2 AND
                MAY BE USED IN CONJUNTION WITH THE PROGRAM LOOPING FEATURES TO
                PROBE THE KD11-K DURING THE FAILING TEST.

        %

        .TITLE  CQKDA-C KD11-K BASIC LOGIC TESTS
        ;*COPYRIGHT (C) 1977,1978
        ;*DIGITAL EQUIPMENT CORP.
        ;*MAYNARD, MASS. 01754
        ;*
        ;*
        ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
        ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
        ;*
        .SBTTL  OPERATIONAL SWITCH SETTINGS
        ;*
        ;*      SWITCH                  USE
        ;*      ------          --------------------
```

```
                        ;*          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 CONTAIN 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
001044  177560         $TKS:    177560                  ;;TTY KBD STATUS
001046  177562         $TKB:    177562                  ;;TTY KBD BUFFER
001050  177564         $TPS:    177564                  ;;TTY PRINTER STATUS REG. ADDRESS
001052  177566         $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  067764                          DT1     ;$REG4,  $REG3,  $REG2, $REG1,$REG0,$ERRPC,$REG5,$REG6
001156  000000                          0
                                ;ITEM 2
001160  064640                          EM2     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001162  065061                          DH2     ;                       IS R3
001164  067764                          DT2     ;$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  070006                          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  067764                          DT4     ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001206  000000                          0
                                ;ITEM 5
001210  064666                          EM5     ;(IR)  TEST  (PC)  (SP)  (PSW)
001212  000000                          0
001214  067772                          DT5     ;$REG1, $REG0, $ERRPC, $REG5, $REG6
001216  000000                          0
                                ;ITEM 6
001220  064660                          EM6     ;  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001222  000000                          0
001224  067770                          DT6     ;$REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001226  000000                          0
                                ;ITEM 7
001230  064650                          EM7     ;WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001232  000000                          0
001234  067766                          DT7     ;$REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001236  000000                          0
                                ;ITEM 10
001240  064724                          EM10    ;S/B RES  WAS RES  DST OP  STC OP  TEST  (PC) (SP) (PSW)
001242  000000                          0
001244  067764                          DT10    ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001246  000000                          0
```

```
1013
1014  001250  065260                ;ITEM 11
1015  001252  065276                        EM11    ;TESTS SKIPPED
1016  001254  070024                        DH11    ;  PC    EXPCTD ACTUAL (TEST #'S)
1017  001256  000000                        DT11    ;$ERRPC, $TESTN,$REG0
1018                                          0
1019                                ;ITEM 12
1020  001260  065334                EM12    ;MED DID NOT ABORT IN USER MODE
1021  001262  067372                DH23    ;PC
1022  001264  067664                DT23    ;$ERRPC
1023  001266  000000                0
1024
1025                                ;ITEM 13
1026  001270  065373                EM13    ;MED EXECUTED IN USER MODE
1027  001272  067372                DH23    ;PC
1028  001274  067664                DT23    ;$ERRPC
1029  001276  000000                0
1030
1031                                ;ITEM 14
1032  001300  065425                EM14    ;MED CHANGED PSW
1033  001302  067372                DH23    ;PC
1034  001304  067664                DT23    ;$ERRPC
1035  001306  000000                0
1036
1037                                ;ITEM 15
1038  001310  065445                EM15    ;MICROBREAK TRAP-TO-4 DID NOT OCCUR
1039  001312  067303                DH15    ;ERRPC MEDCODE MICROBK REG.
1040  001314  067630                DT15    ;$ERRPC,$TMP0,$TMP1,0
1041  001316  067756                DF15    ;0,0
1042
1043                                ;ITEM 16
1044
1045  001320  067034                EM16    ;CACHE DATA LOGGED INCORRECTLY
1046  001322  067606                DH44    ;PC EXPCT RECVD
1047  001324  067670                DT24    ;$ERRPC,$REG1,$REG0,0
1048  001326  000000                0
1049
1050                                ;ITEM 17
1051
1052  001330  067005                EM45    ;CACHE TAG LOGGED WRONG
1053  001332  067606                DH44    ;PC EXPCT RECVD
1054  001334  067670                DT24    ;$ERRPC,$REG0,$REG1,0
1055  001336  000000                0
1056
1057                                ;ITEM 20
1058
1059  001340  065704                EM26    ;PHYS. BA LOGGED WRONG
1060  001342  067606                DH44    ;PC EXPCT RECVD
1061  001344  067670                DT24    ;$ERRPC,$REG1,$REG0,0
1062  001346  000000                0
1063
1064                                ;ITEM 21
1065  001350  065534                EM21    ;CSP CONSTANT WRONG
1066  001352  067335                DH17    ;PC MEDCODE EXPECTD RECEIVD
1067  001354  067640                DT21    ;$ERRPC,$TMP1,$TMP2,$REG0,0
1068  001356  067760                DF17    ;0,0,0
```

```
1069
1070                                    ;ITEM 22
1071  001360  065557                    EM22    ;BAD DATA READ BY A MED
1072  001362  067335                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1073  001364  067652                    DT22    ;$ERRPC,$TMP1,$TMP2,$TMP3,0
1074  001366  067760                    DF17    ;0,0,0
1075
1076                                    ;ITEM 23
1077  001370  065606                    EM23    ;NO ODD PC TRAP
1078  001372  067372                    DH23    ;PC
1079  001374  067664                    DT23    ;$ERRPC
1080  001376  000000                    0
1081
1082                                    ;ITEM 24
1083
1084  001400  065625                    EM24    ;ODD ADR. BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1085  001402  067377                    DH24    ;PC CPUERR LOGJAM
1086  001404  067670                    DT24    ;$ERRPC,$REG1,$REG0
1087  001406  000000                    0
1088
1089                                    ;ITEM 25
1090
1091  001410  065510                    EM17    ;LOG CUA LOGGED INCORRECT U-ADDR
1092  001412  067606                    DH44    ;PC     EXPCTD  RECVD
1093  001414  067670                    DT24    ;$ERRPC $REG1   $REG0
1094  001416  000000                    0
1095
1096                                    ;ITEM 26
1097
1098  001420  065704                    EM26    ;PHYS. BA LOGGED WRONG
1099  001422  067436                    DH26    ;PC PA<17:16>-EXPCT-PA<15:0>  PA<17:16>-RECVD-PA<15:0>
1100  001424  067706                    DT26    ;$ERRPC,$REG1,$REG2,$REG0,$REG3,0
1101  001426  000000                    0
1102
1103                                    ;ITEM 27
1104
1105  001430  065731                    EM27    ;CACHE PARITY ERROR LOGGED IN BACK UP MODE
1106  001432  067520                    DH27    ;PC LOGPBA LOGDATA LOGTAG
1107  001434  067722                    DT27    ;$ERRPC,$REG3,$REG1,$REG2
1108  001436  000000                    0
1109
1110                                    ;ITEM 30
1111
1112  001440  066001                    EM30    ;CACHE PARITY TRAPPED WHEN DISABLED
1113  001442  067372                    DH23    ;PC
1114  001444  067664                    DT23    ;$ERRPC
1115  001446  000000                    0
1116
1117                                    ;ITEM 31
1118
1119  001450  066604                    EM31    ;NO CACHE PARITY TRAP
1120  001452  067372                    DH23    ;PC
1121  001454  067664                    DT23    ;$ERRPC
1122  001456  000000                    0
1123
1124                                    ;ITEM 32
```

```
 1125
 1126  001460  066113                EM32    ;MEMORY ERROR REGISTERS INCORRECT
 1127  001462  067553                DH32    ;PC MEMERR
 1128  001464  067700                DT25    ;$ERRPC,$REG0
 1129  001466  000000                0
 1130
 1131                                ;ITEM 33
 1132
 1133  001470  066144                EM33    ;TIMEOUT BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
 1134  001472  067377                DH24    ;PC CPUERR LOGJAM
 1135  001474  067670                DT24    ;$ERRPC,$REG1,$REG0
 1136  001476  000000                0
 1137
 1138                                ;ITEM 34
 1139
 1140  001500  066222                EM34    ;NO ILLEGAL INTERNAL ADDRESS TRAP
 1141  001502  067372                DH23    ;PC
 1142  001504  067664                DT23    ;$ERRPC
 1143  001506  000000                0
 1144
 1145                                ;ITEM 35
 1146
 1147  001510  066257                EM35    ;INTERNAL ADDRESS ERROR BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
 1148  001512  067377                DH24    ;PC CPUERR LOGJAM
 1149  001514  067670                DT24    ;$ERRPC,$REG1,$REG0
 1150  001516  000000                0
 1151
 1152                                ;ITEM 36
 1153
 1154  001520  066345                EM36    ;LAST INTERRUPT/TRAP VECTOR NOT LOGGED IN FLAG REGISTER
 1155  001522  067422                DH25    ;PC FLGREG
 1156  001524  067700                DT25    ;$ERRPC,$REG0
 1157  001526  000000                0
 1158
 1159                                ;ITEM 37
 1160
 1161  001530  066422                EM37    ;LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR
 1162  001532  067377                DH24    ;PC CPUERR LOGJAM
 1163  001534  067670                DT24    ;$ERRPC,$REG1,$REG0
 1164  001536  000000                0
 1165
 1166                                ;ITEM 40
 1167
 1168  001540  066515                EM40    ;ERROR LOG WAS NOT RE-ENABLED, ODD ADR BIT CLR IN CPUERR
 1169  001542  067377                DH24    ;PC CPUERR LOGJAM
 1170  001544  067670                DT24    ;$ERRPC,$REG1,$REG0
 1171  001546  000000                0
 1172
 1173                                ;ITEM 41
 1174
 1175  001550  066044                EM41    ;INSTRUCTION NOT ABORTED IN CACHE ABORT MODE
 1176  001552  067372                DH23    ;PC
 1177  001554  067664                DT23    ;$ERRPC
 1178  001556  000000                0
 1179
 1180                                ;ITEM 42
```

```
1181
1182  001560  066631                EM42    ;LO BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE
1183  001562  067567                DH42    ;PC LOGSERVCE
1184  001564  067700                DT25    ;$ERRPC,$REG0,0
1185  001566  000000                0
1186
1187                                ;ITEM 43
1188
1189  001570  066717                EM43    ;LO BYTE & TAG PARITY BITS NOT SET IN MEM ERR REGISTER
1190  001572  067553                DH32    ;PC MEMERR
1191  001574  067700                DT25    ;$ERRPC,$REG0
1192  001576  000000                0
1193
1194                                ;ITEM 44
1195
1196  001600  067064                EMEIS1  ;EIS SET COND CODES WRONG
1197  001602  067221                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1198  001604  067734                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1199  001606  000000                0
1200
1201                                ;ITEM 45
1202
1203  001610  067115                EMEIS2  ;EIS GAVE WRONG RESULT
1204  001612  067221                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1205  001614  067734                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1206  001616  000000                0
1207
1208                                ;ITEM 46
1209
1210  001620  067143                EM46    ;AUTO-INCREMENT (DECREMENT) DID NOT OCCUR
1211  001622  067262                DH46    ;  PC     (IR)     TEST
1212  001624  067746                DT46    ;$ERRPC $TMP0   $REG0
1213  001626  000000                0
1214
1215          076600                        MED = 076600
1216          140000                        UM=     140000
1217          177770                        UBREAK= 177770
1218          177744                        MEMERR=177744
1219          177766                        CPUERR=177766
1220          177746                        CCR=177746
1221          000100                        WWP=BIT6
1222          000001                        DPTRP=BIT0
1223          000200                        PABORT=BIT7
1224          000100                        LO=BIT6
1225          000200                        HI=BIT7
1226          000040                        TAG=BIT5
1227
1228                                        .EQUIV  SP,KSP
1229
1230
1231                                ;*      MED OPERATION CODE DEFINITIONS
1232
1233          000226                        WCNSSW=226
1234          000022                        RDWHAMI=022
1235          000222                        WRWHAMI=222
1236          000144                        RDFLAG=144
```

```
1237    000344          WRFLAG=344
1238    000100          RDLJAM=100
1239    000300          WRLJAM=300
1240    000101          RDLSERVICE=101
1241    000301          WRLSERVICE=301
1242    000102          RDLPBA=102
1243    000302          WRLPBA=302
1244    000103          RDLCUA=103
1245    000303          WRLCUA=303
1246    000104          RDLFGINT=104
1247    000304          WRLFGINT=304
1248    000105          RDLWHAMI=105
1249    000305          WRLWHAMI=305
1250    000106          RDLDATA=106
1251    000306          WRLDATA=306
1252    000107          RDLTAG=107
1253    000307          WRLTAG=307
1254    000071          SWB01=71                ;MICRO ADDR. IN SWAB INST.
1255
1256
1257
1258                    ;ADDRESS ASSIGNMENTS FOR DL11 CONSOLE TERMINAL INTERFACE
1259
1260    177560          RCSR=177560             ;RCVR. CONTROL / STATUS REG. ADDRESS
1261    177562          RDBR = 177562           ;RECEIVER DATA BUFFER REG. ADDR.
1262    177564          XCSR = 177564           ;TRANSMITTER CONTROL / STATUS REG. ADDR
1263    177566          XDBR = 177566           ;TRANSMIT DATA BUFFER REG. ADDR.
1264    177546          LKCSR=  177546          ;LINE CLOCK ADDRESS
1265
```

```
                                        ;//////////////////////
                                        ;  "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

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

```
1322
1323
1324
                        .SBTTL BT005 ''CCC AND COND. BR'S'' TEST - FLAGS CLEARED
                ; ******************************************************************************
```

```
1325  001700  000257          BT005:  CCC                     ;MAKE N:C=0000
1326
1327  001702  100403          I005:   BMI     E005            ;BR IF "N" STILL SET
1328  001704  001402                  BEQ     E005            ;BR IF "Z" STILL SET
1329  001706  102401                  BVS     E005            ;BR IF "V" STILL SET
1330  001710  103002                  BCC     BT006           ;BR IF "C" GOT CLEARED
1331
1332  001712  000000          E005:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
1333                                                          ;OR THE CCC FAILED TO CLEAR ALL FLAGS
1334  001714  000771                  BR      BT005           ;LOCK ON HARD ERROR
1335
1336                          ; ***************************************************************************
1337                                  .SBTTL BT006 "CLR  %R" TEST - SETS THE "Z" BIT
1338                          ; ***************************************************************************
1339
1340  001716  000257          BT006:  CCC                     ;MAKE N:C=0000
1341
1342  001720  005000          I006:   CLR     R0              ;TEST THE CLR - IT SHOULD SET "Z"
1343
1344  001722  001402                  BEQ     BT007           ;BR IF CLR SET "Z"
1345
1346  001724  000000          E006:   HALT                    ;ERROR - CLR FAILED TO SET "Z"
1347  001726  000773                  BR      BT006           ;LOCK ON HARD ERROR
1348
1349                          ; ***************************************************************************
1350                                  .SBTTL BT007 "TST  %R" TEST - USING THE CLR
1351                          ; ***************************************************************************
1352
1353  001730  005000          BT007:  CLR     R0              ;MAKE [R0] = 000000
1354  001732  000257                  CCC                     ;MAKE N:C=0000
1355
1356  001734  005700          I007:   TST     R0              ;TEST THE TST - IT SHOULD SET "Z"
1357
1358  001736  001402                  BEQ     BT010           ;BR IF "Z" SET OK
1359
1360  001740  000000          E007:   HALT                    ;ERROR - CLR FAILED TO LOAD R0 WITH
1361                                                          ;ALL ZEROES OR TST FAILED
1362  001742  000772                  BR      BT007           ;LOCK ON HARD ERROR
1363
1364                          ; ***************************************************************************
1365                                  .SBTTL BT010 "COM  %R" TEST - SHOULD SET "N" AND "C"
1366                          ; ***************************************************************************
1367
1368  001744  005000          BT010:  CLR     R0              ;MAKE [R0] = 000000
1369  001746  000257                  CCC                     ;MAKE N:C=0000
1370
1371  001750  005100          I010:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
1372
1373  001752  100001                  BPL     E010            ;BR IF "N" FAILED TO SET
1374  001754  103402                  BCS     BT011           ;BR IF "C" SET OK
1375
1376  001756  000000          E010:   HALT                    ;ERROR - COM FAILED
1377  001760  000771                  BR      BT010           ;LOCK ON HARD ERROR
1378
1379                          ; ***************************************************************************
1380                                  .SBTTL BT011 "COM  %R AND ADC  %R" TEST
```

```
                                ; ***********************************************************************
001762  005000                  BT011:  CLR     R0              ;MAKE [R0] = 000000
001764  000257                          CCC                     ;MAKE N:C=0000

001766  005100                  I011:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
001770  005500                          ADC     R0              ;TEST THE ADC - [R0] S/B = 000000

001772  001001                          BNE     E011            ;BR IF ''Z'' DID NOT SET
001774  103402                          BCS     BT012           ;BR IF ''C'' SET OK

001776  000000                  E011:   HALT                    ;ERROR - COM OR ADC FAILED
002000  000770                          BR      BT011           ;LOCK ON HARD ERROR

                                ; ***********************************************************************
                                        .SBTTL BT012 ''MOV  #N,R'' TEST WITH N=177777,[R]=000000
                                ; ***********************************************************************

002002  005000                  BT012:  CLR     R0              ;MAKE [R0] = 000000
002004  000257                          CCC                     ;MAKE N:C=0000

002006  012700  177777          I012:   MOV     #-1,R0          ;TEST THE MOV - [R0] S/B = 177777

002012  005100                          COM     R0              ;MAKE [R0] = 000000
002014  001402                          BEQ     BT013           ;BR IF ''Z'' SET

002016  000000                  E012:   HALT                    ;ERROR - MOV FAILED TO LOAD R0 WITH ALL 1'S
002020  000770                          BR      BT012           ;LOCK ON HARD ERROR

                                ; ***********************************************************************
                                        .SBTTL BT013 ''MOV  #N,R'' TEST WITH N=000000,[R]=177777
                                ; ***********************************************************************

002022  005000                  BT013:  CLR     R0              ;MAKE [R0] = 000000
002024  005100                          COM     R0              ;MAKE [R0] = 177777
002026  000257                          CCC                     ;SCOPE SYNC

002030  012700  000000          I013:   MOV     #0,R0           ;TEST THE MOV - [R0] S/B = 000000

002034  005100                          COM     R0              ;MAKE [R0] = 177777, SET ''C''
002036  005500                          ADC     R0              ;MAKE [R0] = 000000
002040  001402                          BEQ     BT014           ;BR IF ''Z'' GOT SET

002042  000000                  E013:   HALT                    ;ERROR - MOV FAILED TO CLEAR R0
002044  000766                          BR      BT013           ;LOCK ON HARD ERROR

                                ; ***********************************************************************
                                        .SBTTL BT014 ''CLR  (R)'' TEST - [R] = 177776
                                ; ***********************************************************************

002046  012706  001000          BT014:  MOV     #STACK,SP       ;SET UP STACK POINTER
002052  012700  177776                  MOV     #PSW,R0         ;R0 POINTS TO PSW
002056  000277                          SCC                     ;MAKE [PSW] = 017

002060  005010                  I014:   CLR     (R0)            ;TEST THE CLR - IT SHOULD CLEAR PSW
```

```
                        BNE     BT015           ;BR IF CLR MADE 'Z'' = 0 - IT SHOULD

002064  000000  E014:   HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
002066  000767          BR      BT014           ;LOCK ON HARD ERROR

                ; ******************************************************************************
                        .SBTTL BT015 ''CLR  (R)+'' TEST - [R] = 177776
                ; ******************************************************************************

002070  012700  177776  BT015:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002074  000277          SCC                     ;MAKE [PSW] = 017

002076  005020  I015:   CLR     (R0)+           ;TEST THE CLR - IT SHOULD CLEAR PSW

002100  001002          BNE     A015            ;BR IF CLR MADE 'Z'' = 0 - IT SHOULD

002102  000000  E1015A: HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
002104  000771          BR      BT015           ;LOCK ON HARD ERROR

002106  005700  A015:   TST     R0              ;AUTO INC SHOULD ZERO R0

002110  001402          BEQ     BT016           ;BR IF IT DID

002112  000000  E2015:  HALT                    ;ERROR - AUTOINC. FAILED
002114  000765          BR      BT015           ;LOCK ON HARD ERROR

                ; ******************************************************************************
                        .SBTTL BT016 ''COM  (R)'' TEST - [R] = 177776
                ; ******************************************************************************

002116  012700  177776  BT016:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002122  000257          CCC                     ;MAKE [PSW] = 000

002124  005110  I016:   COM     (R0)            ;TEST THE COM - [PSW] S/B = 357

002126  100003          BPL     E016            ;N:C=1111 ?
002130  001002          BNE     E016
002132  102001          BVC     E016
002134  103403          BCS     BT017

002136  005010  E016:   CLR     (R0)            ;GO TO KERNEL MODE
002140  000000          HALT                    ;ERROR - COM FAILED TO MAKE [PSW] = 357
002142  000765          BR      BT016           ;LOCK ON HARD ERROR

                ; ******************************************************************************
                        .SBTTL BT017 ''COM  (R0)+'' TEST - [R0] = 177776
                ; ******************************************************************************

002144  012700  177776  BT017:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002150  005010          CLR     (R0)            ;MAKE [PSW] = 000
002152  000257          CCC                     ;SCOPE SYNC

002154  005120  I017:   COM     (R0)+           ;TEST THE COM - [PSW] S/B = 357

002156  100003          BPL     EA017           ;N:C = 1111 ?
002160  001002          BNE     EA017
```

```
1493  002162  102001                  BVC     EA017
1494  002164  103405                  BCS     A017
1495
1496  002166  012701  177776  EA017:  MOV     #PSW,R1
1497  002172  005011                  CLR     (R1)
1498  002174  000000                  HALT                    ;COM FAILED TO SET ALL FLAGS
1499  002176  000762                  BR      BT017           ;LOCK ON HARD ERROR
1500
1501  002200  005100          A017:   COM     R0              ;SHOULD MAKE [R0] = 177777
1502  002202  005500                  ADC     R0              ;SHOULD MAKE [R0] = 000000
1503  002204  001405                  BEQ     BT020
1504
1505  002206  012701  177776  E2017:  MOV     #PSW,R1
1506  002212  005011                  CLR     (R1)
1507  002214  000000                  HALT                    ;ERROR - COM FAILED TO AUTO INC. R0
1508  002216  000752                  BR      BT017           ;LOCK ON HARD ERROR
1509
1510                          ; ***************************************************************************
1511                                  .SBTTL BT020 'MOV  RA,RB'' TEST - WITH [RA]=177777,[RB]=000000
1512                          ; ***************************************************************************
1513
1514  002220  012700  177776  BT020:  MOV     #PSW,R0
1515  002224  005010                  CLR     (R0)
1516  002226  005000                  CLR     R0              ;MAKE [R0]=000000
1517  002230  005001                  CLR     R1              ;MAKE [R1]=000000
1518  002232  005101                  COM     R1              ;MAKE [R1]=0207777
1519  002234  000257                  CCC                     ;SCOPE SYNC
1520
1521  002236  010100          I020:   MOV     R1,R0           ;TEST THE MOV.
1522
1523  002240  100402                  BMI     A020            ;BR IF 'N'' GOT SET
1524
1525  002242  000000          EA020:  HALT                    ;ERROR-MOV FAILED TO SET 'N''
1526  002244  000765                  BR      BT020           ;LOCK ON HARD ERROR
1527
1528  002246  005100          A020:   COM     R0              ;[R0] SHOULD GO TO 000000
1529  002250  001402                  BEQ     BT021           ;BR IF IT DID
1530
1531  002252  000000          E2020:  HALT                    ;ERROR-MOV FAILED TO LOAD R0 WITH 1'S
1532  002254  000761                  BR      BT020           ;LOCK ON HARD ERROR
1533
1534                          ; ***************************************************************************
1535                                  .SBTTL BT021 'MOV  RA,RB'' TEST WITH [RA]=000000,[RB]=177777
1536                          ; ***************************************************************************
1537
1538  002256  005000          BT021:  CLR     R0              ;MAKE [R0]=000000
1539  002260  005100                  COM     R0              ;MAKE [R0]=177777
1540  002262  005001                  CLR     R1              ;MAKE [R1]=000000
1541  002264  000257                  CCC                     ;SCOPE SYNC
1542
1543  002266  010100          I021:   MOV     R1,R0           ;TEST THE MOV
1544
1545  002270  001402                  BEQ     A021            ;BR IF ''Z'' GOT SET
1546
1547  002272  000000          EA021:  HALT                    ;MOV FAILED TO SET ''Z''
1548  002274  000770                  BR      BT021           ;LOCK ON HARD ERROR
```

```
1549
1550  002276  005100                  A021:   COM     R0              ;SHOULD MAKE [R0]=177777 AND SET ''C''
1551  002300  005500                          ADC     R0              ;SHOULD MAKE [R0]=000000
1552  002302  001402                          BEQ     BT022           ;BR IF 'Z'' SET
1553
1554  002304  000000                  E2021:  HALT                    ;MOV FAILED TO ZERO R0
1555  002306  000763                          BR      BT021           ;LOCK ON HARD ERROR
1556
1557                                  ; ***************************************************************************
1558                                          .SBTTL BT022 'MOV  #N,@#A'' TEST WITH N=17,A=177776
1559                                  ; ***************************************************************************
1560
1561  002310  000257                  BT022:  CCC                     ;MAKE [PSW]=000
1562
1563  002312  012737  000017  177776  I022:   MOV     #17,@#PSW       ;TEST THE MOV
1564
1565  002320  100003                          BPL     E022            ;N:C=1111
1566  002322  001002                          BNE     E022
1567  002324  102001                          BVC     E022
1568  002326  103402                          BCS     BT023
1569
1570  002330  000000                  E022:   HALT                    ;MOV FAILED TO LOAD PSW
1571  002332  000766                          BR      BT022           ;LOCK ON HARD ERROR
1572
1573                                  ; ***************************************************************************
1574                                          .SBTTL BT023 'MOV  RA,(RB)+'' TEST WITH [RA]=17,[RB]=177776
1575                                  ; ***************************************************************************
1576
1577  002334  012700  177776          BT023:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1578  002340  012701  000017                  MOV     #17,R1          ;[SOURCE]=017
1579  002344  000257                          CCC                     ;SCOPE SYNC - MAKE <N:C> = 0000
1580
1581  002346  010120                  I023:   MOV     R1,(R0)+        ;TEST THE MOV
1582
1583  002350  100003                          BPL     EA023           ;N:C = 1111 ?
1584  002352  001002                          BNE     EA023
1585  002354  102001                          BVC     EA023
1586  002356  103402                          BCS     A023
1587
1588  002360  000000                  EA023:  HALT                    ;MOV FAILED TO LOAD PSW
1589  002362  000764                          BR      BT023           ;LOCK ON HARD ERROR
1590
1591  002364  005700                  A023:   TST     R0              ;DID AUTO INC MAKE R0 GO TO 0?
1592  002366  001402                          BEQ     BT024           ;BR IF IT DID
1593
1594  002370  000000                  E2023:  HALT                    ;MOV FAILED TO AUTO INC. R0
1595  002372  000760                          BR      BT023           ;LOCK ON HARD ERROR
1596
1597                                  ; ***************************************************************************
1598                                          .SBTTL BT024 ''CMP  #N,@#A'' TEST WITH N=(A)
1599                                  ; ***************************************************************************
1600
1601  002374  012700  177776          BT024:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1602  002400  005010                          CLR     (R0)            ;MAKE [PSW]=000
1603  002402  000273                          273                     ;MAKE N:C=1011
1604
```

```
1605  002404  022737  000013  177776  I024:   CMP     #13,@#PSW       ;TEST THE CMP
1606
1607  002412  001402                  BEQ     BT025           ;BR IF ''Z'' GOT SET
1608
1609  002414  000000          E024:   HALT                    ;CMP FAILED TO SET ''Z''
1610  002416  000766                  BR      BT024           ;LOCK ON HARD ERROR
1611
1612                          ; ***************************************************************************
1613                                  .SBTTL BT025 ''CMP  #N,@#A'' WITH N >  (A)
1614                          ; ***************************************************************************
1615
1616  002420  000257          BT025:  CCC                     ;MAKE [PSW]=000
1617
1618  002422  022737  000017  177776  I025:   CMP     #17,@#PSW       ;TEST THE CMP
1619
1620  002430  001401                  BEQ     E025            ;BR IF ''Z'' GOT SET
1621  002432  000402                  BR      BT026           ;GO TO NEXT TEST
1622
1623  002434  000000          E025:   HALT                    ;CMP FAILED TO CLEAR ''Z''
1624  002436  000770                  BR      BT025           ;LOCK ON HARD ERROR
1625                          ; ***************************************************************************
1626                                  .SBTTL BT026 ''CMP  #N,@#A'' WITH N <  (A)
1627                          ; ***************************************************************************
1628
1629  002440  000277          BT026:  SCC                     ;MAKE [PSW]=017
1630
1631  002442  022737  000000  177776  I026:   CMP     #0,@#PSW        ;TEST THE CMP
1632
1633  002450  001401                  BEQ     E026            ;BR IF ''Z'' GOT SET
1634  002452  000402                  BR      BT027           ;GO TO NEXT TEST
1635
1636  002454  000000          E026:   HALT                    ;CMP FAILED TO CLEAR ''Z''
1637  002456  000770                  BR      BT026           ;LOCK ON HARD ERROR
1638
1639                          ; ***************************************************************************
1640                                  .SBTTL BT027 ''CMP  R,#N'' TEST WITH [R]=N
1641                          ; ***************************************************************************
1642
1643  002460  012700  177777  BT027:  MOV     #-1,R0          ;MAKE [R0]=177777
1644  002464  000257                  CCC                     ;N:C=0000
1645
1646  002466  020027  177777  I027:   CMP     R0,#-1          ;TEST THE CMP
1647
1648  002472  001402                  BEQ     BT030           ;BR IF CMP SET ''Z''
1649
1650  002474  000000          E027:   HALT                    ;CMP FAILED
1651  002476  000770                  BR      BT027           ;LOCK ON HARD ERROR
1652
1653                          ; ***************************************************************************
1654                                  .SBTTL BT030 ''CMP  R,#N'' TEST WITH [R] > N
1655                          ; ***************************************************************************
1656
1657  002500  012700  000001  BT030:  MOV     #1,R0           ;MAKE [R0]=000001
1658  002504  000264                  SEZ                     ;SET THE ''Z'' BIT
1659
1660  002506  020027  177777  I030:   CMP     R0,#-1          ;TEST THE CMP
```

```
                                BNE     BT031           ;BR IF CMP CLEARED ''Z''

002514  000000          E030:   HALT                    ;CMP FAILED
002516  000770                  BR      BT030           ;LOCK ON HARD ERROR
                        ; ******************************************************************************
                                .SBTTL BT031 ''CMP  R,#N'' TEST WITH [R] < N
                        ; ******************************************************************************

002520  012700  000001  BT031:  MOV     #1,R0           ;MAKE [R0] = 000001
002524  000264                  SEZ                     ;SET THE ''Z'' BIT

002526  020027  000017  I031:   CMP     R0,#17          ;TEST THE CMP

002532  001002                  BNE     BT032           ;BR IF CMP CLEARED ''Z''

002534  000000          E031:   HALT                    ;CMP FAILED TO SET ''Z''
002536  000770                  BR      BT031           ;LOCK ON HARD ERROR

                        ; ******************************************************************************
                                .SBTTL BT032 ''CMP  (RA)+,RB'' TEST WITH [SOURCE]=[RB]
                        ; ******************************************************************************

002540  012700  177776  BT032:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002544  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW]=340
002552  012701  000340          MOV     #340,R1         ;MAKE [DEST]=340
002556  000257                  CCC                     ;N:C=0000

002560  022001          I032:   CMP     (R0)+,R1        ;TEST THE CMP

002562  001402                  BEQ     A032            ;BR IF ''Z'' GOT SET

002564  000000          EA032:  HALT                    ;CMP FAILED TO ACCESS PSW
002566  000764                  BR      BT032           ;LOCK ON HARD ERROR

002570  005700          A032:   TST     R0              ;''Z'' SHOULD SET
002572  001402                  BEQ     BT033           ;BR IF ''Z'' SET

002574  000000          E2032:  HALT                    ;CMP FAILED TO AUTO INC. R0
002576  000760                  BR      BT032           ;LOCK ON HARD ERROR

                        ; ******************************************************************************
                                .SBTTL BT033 ''CMP  (RA)+,RB'' TEST WITH [SOURCE]>[RB]
                        ; ******************************************************************************

002600  012700  177776  BT033:  MOV     #PSW,R0         ;RC POINTS TO PSW
002604  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW]=340
002612  012701  000330          MOV     #330,R1         ;MAKE [DEST]=330
002616  000264                  SEZ                     ;SET THE ''Z'' BIT

002620  022001          I033:   CMP     (R0)+,R1        ;TEST THE CMP

002622  001002                  BNE     A033            ;BR IF ''Z'' GOT CLEARED

002624  000000          EA033:  HALT                    ;CMP FAILED TO ACCESS PSW
002626  000764                  BR      BT033           ;LOCK ON HARD ERROR
```

```
1717
1718  002630  005700                  A033:   TST     R0              ;"Z" SHOULD SET
1719  002632  001402                          BEQ     BT034           ;BR IF "Z" SET
1720
1721  002634  000000                  E2033:  HALT                    ;CMP FAILED TO AUTO INC. R0
1722  002636  000760                          BR      BT033           ;LOCK ON HARD ERROR
1723                                  ; ******************************************************************************
1724                                          .SBTTL BT034 "CMP  (RA)+,RB" TEST WITH [SOURCE]<[RB]
1725                                  ; ******************************************************************************
1726
1727  002640  012700  177776          BT034:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1728  002644  012737  000330  177776          MOV     #330,@#PSW      ;MAKE [PSW]=330
1729  002652  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
1730  002656  000264                          SEZ                     ;SET THE "Z" BIT
1731
1732  002660  022001                  I034:   CMP     (R0)+,R1        ;TEST THE CMP
1733
1734  002662  001002                          BNE     A034            ;BR IF "Z" GOT CLEARED
1735
1736  002664  000000                  EA034:  HALT                    ;CMP FAILED TO ACCESS PSW
1737  002666  000764                          BR      BT034           ;LOCK ON HARD ERROR
1738
1739  002670  005700                  A034:   TST     R0              ;"Z" SHOULD SET
1740  002672  001402                          BEQ     BT035           ;BR IF "Z" SET
1741
1742  002674  000000                  E2034:  HALT                    ;CMP FAILED TO AUTO INC. R0
1743  002676  000760                          BR      BT034           ;LOCK ON HARD ERROR
1744                                  ; ******************************************************************************
1745                                          .SBTTL BT035 "CMP  RA,RB" TEST WITH [RA] = [RB]
1746                                  ; ******************************************************************************
1747
1748  002700  012700  125252          BT035:  MOV     #125252,R0      ;MAKE [R0] = 125252
1749  002704  010001                          MOV     R0,R1           ;MAKE [R1] = 125252
1750  002706  000257                          CCC                     ;SCOPE SYNC
1751
1752  002710  020100                  I035:   CMP     R1,R0           ;TEST THE CMP
1753
1754  002712  001402                          BEQ     BT036           ;BR IF "Z" GOT SET
1755
1756  002714  000000                  E035:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1757  002716  000770                          BR      BT035           ;LOCK ON HARD ERROR
1758                                  ; ******************************************************************************
1759                                          .SBTTL BT036 "CMP  RA,RB" TEST WITH [RA] < [RB]
1760                                  ; ******************************************************************************
1761
1762  002720  012700  025252          BT036:  MOV     #25252,R0       ;MAKE [R0] = 25252
1763  002724  005001                          CLR     R1              ;MAKE [R1] = 000000
1764  002726  000264                          SEZ                     ;SCOPE SYNC - SET "Z"
1765
1766  002730  020100                  I036:   CMP     R1,R0           ;TEST THE CMP
1767
1768  002732  001002                          BNE     BT037           ;BR IF "Z" GOT  CLEARED
1769
1770  002734  000000                  E036:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1771  002736  000770                          BR      BT036           ;LOCK ON HARD ERROR
1772                                  ; ******************************************************************************
```

```
                                        .SBTTL BT037 ''CMP  RA,RB'' TEST WITH [RA] > [RB]
                                ; ******************************************************************************
002740  005000                BT037:  CLR     R0              ;MAKE [R0] = 000000
002742  012701  000017                MOV     #17,R1          ;MAKE [R1] = 000017
002746  000264                        SEZ                     ;SCOPE SYNC - SET ''Z''

002750  020100                I037:   CMP     R1,R0           ;TEST THE CMP

002752  001002                        BNE     BT040           ;BR IF ''Z'' GOT CLEARED

002754  000000                E037:   HALT                    ;ERROR - CMP FAILED TO SET ''Z''
002756  000770                        BR      BT037           ;LOCK ON HARD ERROR

                                ; ******************************************************************************
                                        .SBTTL BT040 ''MOV  (RA),RB'' TEST WITH [SOURCE]=[RB]=17
                                ; ******************************************************************************

002760  012700  177776        BT040:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002764  005010                        CLR     (R0)            ;MAKE [PSW]=000
002766  005001                        CLR     R1              ;MAKE [R1]=000000
002770  000277                        SCC                     ;MAKE N:C=1111

002772  011001                I040:   MOV     (R0),R1         ;TEST THE MOV

002774  020127  000017                CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
003000  001402                        BEQ     BT041           ;BR IF YES

003002  000000                E040:   HALT                    ;MOV FAILED TO LOAD R1
003004  000765                        BR      BT040           ;LOCK ON HARD ERROR
                                ; ******************************************************************************
                                        .SBTTL BT041 ''MOV  (RA)+,RB'' TEST WITH [SOURCE]=[RB]=17
                                ; ******************************************************************************

003006  012700  177776        BT041:  MOV     #PSW,R0         ;R0 POINTS TO PSW
003012  005010                        CLR     (R0)            ;MAKE [PSW]=000
003014  005001                        CLR     R1              ;MAKE [R1]=000000
003016  000277                        SCC                     ;MAKE N:C=1111

003020  012001                I041:   MOV     (R0)+,R1                ;TEST THE MOV

003022  020127  000017                CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
003026  001402                        BEQ     A041            ;BR IF YES

003030  000000                EA041:  HALT                    ;MOV FAILED TO LOAD R1
003032  000765                        BR      BT041           ;LOCK ON HARD ERROR

003034  005700                A041:   TST     R0              ;''Z'' SHOULD SET
003036  001402                        BEQ     BT042           ;BR IF ''Z'' GOT SET

003040  000000                E2041:  HALT                    ;MOV FAILED TO AUTO INC. R0
003042  000761                        BR      BT041           ;LOCK ON HARD ERROR

                                ; ******************************************************************************
                                        .SBTTL BT042 ''XOR  RA,RB'' TEST WITH [RA] = [RB] = 000000
                                ; ******************************************************************************
```

```
1829
1830  003044  005000                BT042:  CLR     R0              ;MAKE [R0] = 000000
1831  003046  005001                        CLR     R1              ;MAKE [R1] = 000000
1832  003050  000257                        CCC                     ;SCOPE SYNC
1833
1834  003052  074100                I042:   XOR     R1,R0           ;TEST THE XOR
1835
1836  003054  005700                        TST     R0              ;RESULT = 000000 ?
1837  003056  001402                        BEQ     BT043           ;BR IF YES
1838
1839  003060  000000                E042:   HALT                    ;XOR FAILED
1840  003062  000770                        BR      BT042
1841
1842                                ; ***************************************************************************
1843                                        .SBTTL BT043 'XOR  RA,RB' TEST WITH [RA] = [RB] = 177777
1844                                ; ***************************************************************************
1845
1846  003064  005000                BT043:  CLR     R0              ;MAKE [R0] = 177777
1847  003066  005100                        COM     R0
1848  003070  010001                        MOV     R0,R1           ;MAKE [R1] = 177777
1849  003072  000257                        CCC                     ;SCOPE SYNC
1850
1851  003074  074100                I043:   XOR     R1,R0           ;TEST THE XOR
1852
1853  003076  005700                        TST     R0              ;RESULT = 000000 ?
1854  003100  001402                        BEQ     BT044   ;BR IF YES
1855
1856  003102  000000                E043:   HALT                    ;XOR FAILED
1857  003104  000767                        BR      BT043           ;LOCK ON HARD ERROR
1858
1859                                ; ***************************************************************************
1860                                        .SBTTL BT044 'XOR  RA,RB' TEST WITH [RB]=052525,[RA]=125252
1861                                ; ***************************************************************************
1862
1863  003106  012701  125252        BT044:  MOV     #125252,R1      ;MAKE [R1]=125252
1864  003112  012700  052525                MOV     #052525,R0      ;MAKE [R0]=052525
1865  003116  000257                        CCC                     ;SCOPE SYNC
1866
1867  003120  074100                I044:   XOR     R1,R0           ;TEST THE XOR
1868
1869  003122  020027  177777                CMP     R0,#-1          ;RESULT = 177777 ?
1870  003126  001402                        BEQ     BT045           ;BR IF YES
1871
1872  003130  000000                E044:   HALT                    ;XOR FAILED
1873  003132  000400                        BR      BT045           ;LOCK ON HARD ERROR
1874                                ; ***************************************************************************
1875                                        .SBTTL BT045 'XOR  RA,RB' TEST WITH [RA]=052525,[RB]=125252
1876                                ; ***************************************************************************
1877
1878  003134  012700  125252        BT045:  MOV     #125252,R0      ;MAKE [R0]=125252
1879  003140  012701  052525                MOV     #052525,R1      ;MAKE [R1]=052525
1880  003144  000257                        CCC                     ;SCOPE SYNC
1881
1882  003146  074100                I045:   XOR     R1,R0           ;TEST THE XOR
1883
1884  003150  020027  177777                CMP     R0,#-1          ;RESULT = 177777 ?
```

```
1885  003154  001402                  BEQ     BT046           ;BR IF YES
1886
1887  003156  000000          E045:   HALT                    ;XOR FAILED
1888  003160  000765                  BR      BT045           ;LOCK ON HARD ERROR
1889
1890                          ; ***************************************************************************
1891                                  .SBTTL BT046 GPR ADDRESS INTERRACTION TEST
1892                          ; ***************************************************************************
1893
1894  003162  012700  125252  BT046:  MOV     #125252,R0      ;[R0] = 125252
1895  003166  010001                  MOV     R0,R1
1896  003170  005101                  COM     R1              ;[R1] = 052525
1897  003172  010102                  MOV     R1,R2
1898  003174  005102                  COM     R2              ;[R2] = 125252
1899  003176  010203                  MOV     R2,R3
1900  003200  005103                  COM     R3              ;[R3] = 052525
1901  003202  010304                  MOV     R3,R4
1902  003204  005104                  COM     R4              ;[R4] = 125252
1903  003206  010405                  MOV     R4,R5
1904  003210  005105                  COM     R5              ;[R5] = 052525
1905
1906  003212  074100          I046:   XOR     R1,R0           ;[R0] S/B = 177777
1907  003214  074200                  XOR     R2,R0           ;[R0] S/B = 125252
1908  003216  074300                  XOR     R3,R0           ;[R0] S/B = 177777
1909  003220  074400                  XOR     R4,R0           ;[R0]  S/B = 125252
1910  003222  074500                  XOR     R5,R0           ;[R0]  S/B = 177777
1911  003224  005100                  COM     R0              ;[R0] S/B = 000000
1912
1913  003226  001402                  BEQ     A046            ;BR IF [R0] WAS 000000
1914
1915  003230  000000          EA046:  HALT                    ;GPR ADDRESSING PROBLEM
1916  003232  000753                  BR      BT046           ;LOCK ON HARD ERROR
1917
1918  003234  020627  001000  A046:   CMP     SP,#STACK       ;DID R6 GET DISTURBED
1919  003240  001402                  BEQ     BASIC           ;BR IF NOT
1920
1921  003242  000000          E2046:  HALT                    ;R6 ADDRESS PROBLEM
1922  003244  000746                  BR      BT046           ;LOCK ON HARD ERROR
```

```
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                ;          /  /  /  /  /  / BASIC INSTRUCTION TESTS /  /  /  /  /  /
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

003246  005037  063254        BASIC:  CLR     @#ONCE          ;SIGNAL PROGRAM HEADER TO BE PRINTED
003252  005037  001012                CLR     @#$ERTTL        ;CLEAR ERROR COUNT FIRST TIME THROUGH
003256  005037  001126                CLR     @#$PASS         ;CLEAR PASS COUNT FIRST TIME THROUGH
003262  012701  063236        INIT:   MOV     #PRIFLG,R1      ;SET UP TO INIT. COUNTERS AND FLAGS
003266  005021                1$:     CLR     (R1)+           ;CLEAR ONE WORD
003270  020127  063254                CMP     R1,#ONCE        ;CLEARED ALL FLAGS AND COUNTERS?
003274  001374                        BNE     1$              ;BR IF NOT
003276  012706  001000                MOV     #STACK,SP       ;SET UP THE STACK POINTER
003302  012737  004030  177770        MOV     #4030,@#UBREAK  ;SET SCOPE SYNC FOR COND CODE OPERATE
003310  012737  177777  001074        MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
                                                              ;IN FIRST ERROR CALL

                                ;;**********************************************************************
                                ;*TEST 0         BASIC 'BNE'' TEST WITH Z=0
                                ;;**********************************************************************
003316                          TST0:
003316  012700  000000                MOV     #0,R0           ;;LOAD R0 WITH TEST NUMBER
003322  000257                1$:     CCC                     ;MAKE Z=0

003324                          2$:
003324  001002                        BNE     TST1            ;;TEST THE BNE - IT SHOULD BR

003326  000000                3$:     HALT                    ;BNE FAILED TO LOAD PC
003330  000774                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 1         BASIC 'BNE'' TEST WITH Z=1
                                ;;**********************************************************************
003332                          TST1:
003332  012700  000001                MOV     #1,R0           ;;LOAD R0 WITH TEST NUMBER
003336  000264                1$:     SEZ                     ;SET THE 'Z'' BIT

003340  001001                2$:     BNE     3$              ;TEST THE BNE - IT SHOULD NOT BR

003342  000402                        BR      TST2            ;;GO TO NEXT TEST

003344  000000                3$:     HALT                    ;BNE BRANCHED WITH Z=1
003346  000773                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 2         BASIC 'BEQ'' TEST WITH Z=1
                                ;;**********************************************************************
003350                          TST2:
003350  012700  000002                MOV     #2,R0           ;;LOAD R0 WITH TEST NUMBER
003354  000264                1$:     SEZ                     ;MAKE Z=1

003356                          2$:
003356  001402                        BEQ     TST3            ;;TEST THE BEQ - IT SHOULD BR

003360  000000                3$:     HALT                    ;BEQ FAILED TO LOAD THE PC
003362  000774                        BR      1$              ;LOCK ON HARD ERROR
                                ;;**********************************************************************
```

```
1979
1980
1981  003364
1982  003364  012700  000003          ;*TEST 3        BASIC "BEQ" TEST WITH Z=0
1983  003370  000257                  ;;**************************************************************
1984                                  TST3:
1985  003372  001401                          MOV     #3,R0           ;;LOAD R0 WITH TEST NUMBER
1986                                  1$:     CCC                     ;MAKE Z=0
1987  003374  000402
1988                                  2$:     BEQ     3$              ;TEST THE BEQ - IT SHOULD NOT BR
1989  003376  000000                          BR      TST4            ;;GO TO NEXT TEST
1990  003400  000773
1991                                  3$:     HALT                    ;BEQ BRANCHED WITH Z=0
1992                                          BR      1$              ;LOCK ON HARD ERROR
1993
1994                                  ;;**************************************************************
1995  003402                          ;*TEST 4        BASIC "BPL" TEST WITH N=1
1996  003402  012700  000004          ;;**************************************************************
1997  003406  005037  177776          TST4:
1998  003412  000270                          MOV     #4,R0           ;;LOAD R0 WITH TEST NUMBER
1999                                  1$:     CLR     @#PSW           ;CLEAR THE PSW
2000  003414  100001                          SEN                     ;MAKE N=1
2001
2002  003416  000402                  2$:     BPL     3$              ;TEST THE BPL - IT SHOULDN'T BR
2003
2004  003420  000000                          BR      TST5            ;;GO TO NEXT TEST
2005  003422  000771
2006                                  3$:     HALT                    ;BPL BRANCHED WITH N=1
2007                                          BR      1$              ;LOCK ON HARD ERROR
2008
2009                                  ;;**************************************************************
2010  003424                          ;*TEST 5        BASIC "BPL" TEST WITH N=0
2011  003424  012700  000005          ;;**************************************************************
2012  003430  005037  177776          TST5:
2013  003434  000257                          MOV     #5,R0           ;;LOAD R0 WITH TEST NUMBER
2014                                  1$:     CLR     @#PSW           ;CLEAR THE PSW
2015  003436                                  CCC                     ;SCOPE SYNC
2016  003436  100002
2017                                  2$:
2018  003440  000000                          BPL     TST6            ;;TEST THE BPL - IT SHOULD BR
2019  003442  000772
2020                                  3$:     HALT                    ;BPL FAILED TO LOAD THE PC
2021                                          BR      1$              ;LOCK ON HARD ERROR
2022
2023                                  ;;**************************************************************
2024  003444                          ;*TEST 6        BASIC "MOV  (RA),RB" TEST - (RA)=177776
2025  003444  012700  000006          ;;**************************************************************
2026  003450  012705  177776          TST6:
2027  003454  005015                          MOV     #6,R0           ;;LOAD R0 WITH TEST NUMBER
2028  003456  005003                          MOV     #PSW,R5         ;SOURCE ADDR = 177776
2029  003460  000277                  1$:     CLR     (R5)            ;MAKE [PSW]=000
2030                                          CLR     R3              ;[DEST] = 000000
2031  003462  011503                          SCC                     ;MAKE [PSW]=017
2032
2033  003464  020327  000017          2$:     MOV     (R5),R3         ;TEST THE MOV
2034  003470  001402
                                              CMP     R3,#17          ;CORRECT RESULT ?
                                              BEQ     TST7            ;,BR IF YES
```

```
2035
2036  003472  000000                  3$:     HALT                    ;ERROR-MOV FAILED
2037  003474  000767                          BR      1$              ;LOCK ON HARD ERROR
2038                                  ;;**********************************************************************
2039                                  ;*TEST 7         BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
2040                                  ;;**********************************************************************
2041  003476                          TST7:
2042  003476  012700  000007                 MOV     #7,R0           ;;LOAD R0 WITH TEST NUMBER
2043  003502  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2044  003506  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2045  003512  012737  125252  063312 1$:     MOV     #125252,@#MBUF0 ;MAKE [DEST] = 125252
2046  003520  000257                         CCC                     ;MAKE N:C=0000
2047
2048  003522  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2049
2050  003524  001402                          BEQ     TST10           ;;BR IF "Z" GOT SET
2051
2052  003526  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO SET "Z"
2053  003530  000770                          BR      1$              ;LOCK ON HARD ERROR
2054                                  ;;**********************************************************************
2055                                  ;*TEST 10        BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
2056                                  ;;**********************************************************************
2057  003532                          TST10:
2058  003532  012700  000010                 MOV     #10,R0          ;;LOAD R0 WITH TEST NUMBER
2059  003536  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2060  003542  012704  000001                 MOV     #1,R4           ;RESULT S / B = 000001
2061  003546  005037  063312          1$:    CLR     @#MBUF0         ;MAKE [DEST] = 000000
2062  003552  000264                         SEZ                     ;MAKE N:C=0100
2063
2064  003554  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
2065
2066  003556  001002                          BNE     TST11           ;;BR IF "Z" GOT CLEARED
2067
2068  003560  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO CLR "Z"
2069  003562  000771                          BR      1$              ;LOCK ON HARD ERROR
2070
2071                                  ;;**********************************************************************
2072                                  ;*TEST 11        BASIC "CMP #N,R" TEST - N = [R]
2073                                  ;;**********************************************************************
2074  003564                          TST11:
2075  003564  012700  000011                 MOV     #11,R0          ;;LOAD R0 WITH TEST NUMBER
2076  003570  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2077  003574  010403                  1$:    MOV     R4,R3           ;[DEST] = 125252
2078  003576  000257                         CCC                     ;SCOPE SYNC
2079
2080  003600  022703  125252          2$:    CMP     #125252,R3      ;TEST THE CMP
2081
2082  003604  001402                          BEQ     4$              ;BR IF N = [R]
2083
2084  003606  000000                  3$:     HALT                    ;CMP FAILED
2085  003610  000771                          BR      1$              ;LOCK ON HARD ERROR
2086
2087  003612  020403                  4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
2088  003614  001402                          BEQ     TST12           ;;BR IF NO
2089
2090  003616  000000                  5$:     HALT                    ;CMP DELIVERED A RESULT
```

```
2091  003620  000765                    BR      1$              ;LOCK ON HARD ERROR
2092
2093                                    ;;*****************************************************************
2094                                    ;*TEST 12        BASIC ''CMP #N,R'' TEST - N NOT EQUAL TO [R]
2095                                    ;;*****************************************************************
2096  003622                            TST12:
2097  003622  012700  000012                   MOV     #12,R0          ;;LOAD R0 WITH TEST NUMBER
2098  003626  005004                           CLR     R4              ;RESULT S / B = 000000
2099  003630  010403                    1$:    MOV     R4,R3           ;[DEST] = 125252
2100  003632  000264                           SEZ                     ;SCOPE SYNC
2101
2102  003634  022703  000001            2$:    CMP     #1,R3           ;TEST THE CMP
2103
2104  003640  001002                           BNE     4$              ;BR IF N NOT EQUAL TO [R]
2105
2106  003642  000000                    3$:    HALT                    ;CMP FAILED
2107  003644  000771                           BR      1$              ;LOCK ON HARD ERROR
2108
2109  003646  020403                    4$:    CMP     R4,R3           ;DID CMP ALTER [DEST]?
2110  003650  001402                           BEQ     TST13           ;;BR IF NO
2111
2112  003652  000000                    5$:    HALT                    ;CMP DELIVERED A RESULT
2113  003654  000765                           BR      1$              ;LOCK ON HARD ERROR
2114
2115                                    ;;*****************************************************************
2116                                    ;*TEST 13        BASIC 'MOV  RA,(RB)'' TEST
2117                                    ;;*****************************************************************
2118  003656                            TST13:
2119  003656  012700  000013                   MOV     #13,R0          ;;LOAD R0 WITH TEST NUMBER
2120  003662  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR=MBUF0
2121  003666  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
2122  003672  005012                    1$:    CLR     (R2)            ;MAKE [DEST] = 000000
2123  003674  000257                           CCC                     ;SCOPE SYNC - N:C=0000
2124
2125  003676  010412                    2$:    MOV     R4,(R2)         ;TEST THE MOV
2126
2127  003700  020412                           CMP     R4,(R2)         ;RESULT CORRECT ?
2128  003702  001402                           BEQ     TST14           ;;BR IF  YES
2129
2130  003704  000000                    3$:    HALT                    ;ERROR - MOV FAILED
2131  003706  000771                           BR      1$              ;LOCK ON HARD ERROR
2132
2133                                    ;;*****************************************************************
2134                                    ;*TEST 14        BASIC 'MOV  #N,(R)'' TEST
2135                                    ;;*****************************************************************
2136  003710                            TST14:
2137  003710  012700  000014                   MOV     #14,R0          ;;LOAD R0 WITH TEST NUMBER
2138  003714  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2139  003720  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
2140  003724  005012                    1$:    CLR     (R2)            ;MAKE [DEST] = 000000
2141  003726  000257                           CCC                     ;SCOPE SYNC
2142
2143  003730  012712  177777            2$:    MOV     #-1,(R2)        ;TEST THE MOV
2144
2145  003734  020412                           CMP     R4,(R2)         ;RESULT OK ?
2146  003736  001402                           BEQ     TST15           ;;BR IF YES
```

```
2147
2148  003740  000000                  3$:     HALT                    ;ERROR - MOV FAILED
2149  003742  000770                          BR      1$              ;LOCK ON HARD ERROR
2150
2151                                  ;;**********************************************************************
2152                                  ;*TEST 15       BASIC 'MOVB #N,X(R)' TEST - DEST EVEN
2153                                  ;;**********************************************************************
2154  003744                          TST15:
2155  003744  012700  000015                 MOV     #15,R0          ;;LOAD R0 WITH TEST NUMBER
2156  003750  012704  177401                 MOV     #177401,R4      ;RESULT S / B = 177401
2157  003754  012702  063316                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2158  003760  012705  063312                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2159  003764  012712  177777         1$:     MOV     #-1,(R2)        ;[DEST] = 177777
2160  003770  000257                         CCC                     ;SCOPE SYNC
2161
2162  003772  112765  000001  000004 2$:     MOVB    #1,4(R5)        ;TEST THE MOVB
2163
2164  004000  020412                         CMP     R4,(R2)         ;RESULT OK?
2165  004002  001402                         BEQ     TST16           ;;BR IF YES
2166
2167  004004  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2168  004006  000766                          BR      1$              ;LOCK ON HARD ERROR
2169
2170                                  ;;**********************************************************************
2171                                  ;*TEST 16       BASIC 'MOVB #N,X(R)' TEST - DEST ODD
2172                                  ;;**********************************************************************
2173  004010                          TST16:
2174  004010  012700  000016                 MOV     #16,R0          ;;LOAD R0 WITH TEST NUMBER
2175  004014  012704  000777                 MOV     #777,R4         ;RESULT S / B = 777
2176  004020  012702  063316                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2177  004024  012705  063312                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2178  004030  012712  177777         1$:     MOV     #-1,(R2)        ;[DEST] = 177777
2179  004034  000257                         CCC                     ;SCOPE SYNC
2180
2181  004036  112765  000001  000005 2$:     MOVB    #1,5(R5)        ;TEST THE MOVB
2182
2183  004044  020412                         CMP     R4,(R2)         ;RESULT OK?
2184  004046  001402                         BEQ     TST17           ;;BR IF YES
2185
2186  004050  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2187  004052  000766                          BR      1$              ;LOCK ON HARD ERROR
2188
2189                                  ;;**********************************************************************
2190                                  ;*TEST 17       BASIC 'TST  @#A' TEST WITH [A] GT 0
2191                                  ;;**********************************************************************
2192  004054                          TST17:
2193  004054  012700  000017                 MOV     #17,R0          ;;LOAD R0 WITH TEST NUMBER
2194  004060  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2195  004064  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377 (NO CHANGE)
2196  004070  010412         1$:     MOV     R4,(R2)         ;[DEST] = 377
2197  004072  000257                         CCC                     ;SCOPE SYNC
2198
2199  004074  005737  063312         2$:     TST     @#MBUF0         ;TEST THE TST
2200
2201  004100  001401                         BEQ     3$              ;BR IF 'Z' SET - IT SHOULDN'T BE
2202  004102  100002                         BPL     TST20           ;;BR IF 'N' CLEAR - IT SHOULD BE
```

```
                                3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
                                        BR      1$              ;LOCK ON HARD ERROR
                                ;;**********************************************************************
                                ;*TEST 20        BASIC "TST  @#A" TEST WITH [A] LT 0
                                ;;**********************************************************************
                                TST20:
                                        MOV     #20,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #100000,R4      ;MAKE S / B = 100000
                                1$:     MOV     R4,(R2)         ;MAKE [DEST] = 100000
                                        CCC                     ;SCOPE SYNC

                                2$:     TST     @#MBUF0         ;TEST THE TST

                                        BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
                                        BMI     4$              ;BR IF "N" SET - IT SHOULD BE

                                3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
                                        BR      1$              ;LOCK ON HARD ERROR
                                4$:     CMP     R4,(R2)         ;DID TST DISTURB [DEST] ?
                                        BEQ     TST21           ;;BR IF NOT

                                5$:     HALT                    ;TST DELIVERED A RESULT
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 21        BASIC "TST  @#A" WITH [A] = 0
                                ;;**********************************************************************
                                TST21:
                                        MOV     #21,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        CLR     R4              ;RESULT S / B = 0 (IT SHOULDN'T CHANGE
                                1$:     CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC - Z=0

                                2$:     TST     @#MBUF0         ;TEST THE TST

                                        BEQ     4$              ;BR IF TST SET "Z"

                                3$:     HALT                    ;TST FAILED TO SET "Z"
                                        BR      1$              ;LOCK ON HARD ERROR

                                4$:     CMP     R4,(R2)         ;[DEST] STILL = 000000
                                        BEQ     TST22           ;;BR IF YES

                                5$:     HALT                    ;TST ALTERED THE [DEST]
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 22        BASIC "BIT  #N,@#A" WITH BIT SET IN "A"
                                ;;**********************************************************************
                                TST22:
                                        MOV     #22,R0          ;;LOAD R0 WITH TEST NUMBER
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #40000,R4       ;RESULT S / B = 40000
```

```
004230  010412                  1$:     MOV     R4,(R2)          ;MAKE [DEST] = 40000
004232  000277                          SCC                      ;SCOPE SYNC - Z=1

004234  032737  040000  063312  2$:     BIT     #40000,@#MBUF0   ;TEST THE BIT

004242  001002                          BNE     TST23            ;;BR IF Z=0 - IT SHOULD BE

004244  000000                  3$:     HALT                     ;BIT FAILED TO CLEAR "Z"
004246  000770                          BR      1$               ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 23        BASIC "BIT  #N,@#A" WITH BIT CLEAR IN "A"
                                ;;*******************************************************************
004250                          TST23:
004250  012700  000023                  MOV     #23,R0           ;;LOAD R0 WITH TEST NUMBER
004254  012702  063312                  MOV     #MBUF0,R2        ;DEST ADDR = MBUF0
004260  005012                  1$:     CLR     (R2)             ;MAKE [DEST] = 000000
004262  000257                          CCC                      ;SCOPE SYNC - Z=0

004264  032737  040000  063312  2$:     BIT     #40000,@#MBUF0   ;TEST THE BIT

004272  001402                          BEQ     4$               ;BR IF Z=1 - IT SHOULD BE

004274  000000                  3$:     HALT                     ;BIT FAILED TO SET "Z"
004276  000770                          BR      1$               ;LOCK ON HARD ERROR

004300  005712                  4$:     TST     (R2)             ;DID BIT DELIVER A RESULT
004302  001402                          BEQ     TST24            ;;BR IF NOT

004304  000000                  5$:     HALT                     ;BIT DISTURBED THE [DEST]
004306  000764                          BR      1$               ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 24        BASIC "TST  (R)+" TEST
                                ;;*******************************************************************
004310                          TST24:
004310  012700  000024                  MOV     #24,R0           ;;LOAD R0 WITH TEST NUMBER
                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT0
004314  032737  000001  063234          BIT     #BIT0,@#BPTLOC   ;BREAKPOINT HALT SET ??
004322  001401                          BEQ     .+4              ;BR IF NOT
004324  000000                          HALT                     ;BREAK - DEPRESS CONTINUE TO RESTART
004326  012702  063312                  MOV     #MBUF0,R2        ;INITIAL DEST ADDR = MBUF0
004332  005012                  1$:     CLR     (R2)             ;MAKE [DEST] = 000000
004334  000257                          CCC                      ;SCOPE SYNC

004336  005722                  2$:     TST     (R2)+            ;TEST THE TST

004340  001402                          BEQ     4$               ;BR IF "Z" SET - IT SHOULD BE

004342  000000                  3$:     HALT                     ;TST FAILED TO SET "Z"
004344  000772                          BR      1$               ;LOCK ON HARD ERROR

004346  022702  063314          4$:     CMP     #MBUF0+2,R2      ;DID REG. GET AUTO-INCREMENTED ?
004352  001402                          BEQ     TST25            ;;BR IF YES

004354  000000                  5$:     HALT                     ;TST FAILED TO UPDATE REGISTER
```

```
2315  004356  000765                    BR      1$              ;LOCK ON HARD ERROR
2316
2317                                    ;;*****************************************************************
2318                                    ;*TEST 25        BASIC ''TST -(R)'' TEST
2319                                    ;;*****************************************************************
2320  004360                            TST25:
2321  004360  012700  000025            MOV     #25,R0          ;;LOAD R0 WITH TEST NUMBER
2322  004364  012702  063330            MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2323  004370  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
2324  004374  012705  063332    1$:     MOV     #DWTA+10,R5     ;BASE DEST ADDR = DWTA+10
2325  004400  000270                    SEN                     ;SCOPE SYNC
2326
2327  004402  005745            2$:     TST     -(R5)           ;TEST THE TST
2328
2329  004404  100002                    BPL     4$              ;BR IF 'N'' CLEAR
2330
2331  004406  000000            3$:     HALT                    ;TST FAILED TO CLEAR 'N''
2332  004410  000771                    BR      1$              ;LOCK ON HARD ERROR
2333
2334  004412  020502            4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED?
2335  004414  001402                    BEQ     6$              ;BR IF YES
2336
2337  004416  000000            5$:     HALT                    ;ERROR - TST FAILED TO UPDATE DEST REG
2338  004420  000765                    BR      1$              ;LOCK ON HARD ERROR
2339
2340  004422  020412            6$:     CMP     R4,(R2)         ;DID TST ALTER [DEST]?
2341  004424  001403                    BEQ     TST26           ;;BR IF NOT
2342
2343  004426  000000            7$:     HALT                    ;TST ALTERED [DEST]
2344  004430  010412                    MOV     R4,(R2)         ;RESTORE [DEST]
2345  004432  000760                    BR      1$              ;LOCK ON HARD ERROR
2346
2347                                    ;;*****************************************************************
2348                                    ;*TEST 26        BASIC ''COM  @#A'' TEST
2349                                    ;;*****************************************************************
2350  004434                            TST26:
2351  004434  012700  000026            MOV     #26,R0          ;;LOAD R0 WITH TEST NUMBER
2352  004440  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2353  004444  005004                    CLR     R4              ;RESULT S / B = 177777
2354  004446  005104                    COM     R4
2355  004450  005012            1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2356  004452  000257                    CCC                     ;SCOPE SYNC
2357
2358  004454  005137  063312    2$:     COM     @#MBUF0 ;TEST THE COM
2359
2360  004460  020412                    CMP     R4,(R2)         ;RESULT = 177777 ??
2361  004462  001402                    BEQ     TST27           ;;BR IF YES
2362
2363  004464  000000            3$:     HALT                    ;COM DELIVERED THE WRONG RESULT
2364  004466  000770                    BR      1$
2365
2366                                    ;;*****************************************************************
2367                                    ;*TEST 27        BASIC ''INC  @#A'' TEST
2368                                    ;;*****************************************************************
2369  004470                            TST27:
2370  004470  012700  000027            MOV     #27,R0          ;.LOAD R0 WITH TEST NUMBER
```

```
2371  004474  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2372  004500  012704  000100          MOV     #100,R4         ;RESULT S / B = 100
2373  004504  012712  000077  1$:     MOV     #77,(R2)        ;[DEST] = 77
2374  004510  000257                  CCC                     ;SCOPE SYNC
2375
2376  004512  005237  063312  2$:     INC     @#MBUF0         ;TEST THE INC
2377
2378  004516  020412                  CMP     R4,(R2)         ;DID RESULT = 100 ??
2379  004520  001402                  BEQ     TST30           ;;BR IF YES
2380
2381  004522  000000  3$:     HALT                            ;INC DELIVERED WRONG RESULT
2382  004524  000767                  BR      1$              ;LOCK ON HARD ERROR
2383
2384                          ;;**************************************************************
2385                          ;*TEST 30        BASIC 'DEC  RN'' TEST
2386                          ;;**************************************************************
2387  004526                  TST30:
2388  004526  012700  000030          MOV     #30,R0          ;;LOAD R0 WITH TEST NUMBER
2389  004532  012703  000001  1$:     MOV     #1,R3           ;[DEST] = +1
2390  004536  000257                  CCC                     ;SCOPE SYNC
2391
2392  004540  005303  2$:     DEC     R3                      ;TEST THE DEC
2393
2394  004542  005703                  TST     R3              ;RESULT = 000000 ??
2395  004544  001402                  BEQ     TST31           ;;BR IF YES
2396
2397  004546  000000  3$:     HALT                            ;DEC DELIVERED THE WRONG RESULT
2398  004550  000770                  BR      1$              ;LOCK ON HARD ERROR
2399
2400                          ;;**************************************************************
2401                          ;*TEST 31        BASIC 'DEC  @#A'' TEST
2402                          ;;**************************************************************
2403  004552                  TST31:
2404  004552  012700  000031          MOV     #31,R0          ;;LOAD R0 WITH TEST NUMBER
2405  004556  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
2406  004562  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2407  004566  005012  1$:     CLR     (R2)                    ;MAKE [DEST] = 000000
2408  004570  000257                  CCC                     ;SCOPE SYNC
2409
2410  004572  005337  063312  2$:     DEC     @#MBUF0         ;TEST THE DEC
2411
2412  004576  020412                  CMP     R4,(R2)         ;DID RESULT = 177777 ??
2413  004600  001402                  BEQ     TST32           ;;BR IF YES
2414
2415  004602  000000  3$:     HALT                            ;DEC DELIVERED WRONG RESULT
2416  004604  000770                  BR      1$              ;LOCK ON HARD ERROR
2417
2418                          ;;**************************************************************
2419                          ;*TEST 32        BASIC ''CLR X(R)'' TESTS
2420                          ;;**************************************************************
2421  004606                  TST32:
2422  004606  012700  000032          MOV     #32,R0          ;;LOAD R0 WITH TEST NUMBER
2423  004612  012702  063314          MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
2424  004616  005004                  CLR     R4              ;RESULT S / B = 000000
2425  004620  012705  063312  1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2426  004624  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
```

```
 2427  004630  000257                    CCC                     ;SCOPE SYNC
 2428
 2429  004632  005065  000002  2$:       CLR     2(R5)           ;TEST THE CLR
 2430
 2431  004636  020412                    CMP     R4,(R2)         ;RESULT = 0?
 2432  004640  001402                    BEQ     TST33           ;;BR IF YES
 2433
```

```
2434  004642  000000                3$:   HALT                    ;CLR FAILED TO ZERO [DEST]
2435  004644  000765                      BR    1$                ;LOCK ON HARD ERROR.
2436
2437                                ;;*********************************************************************
2438                                ;*TEST 33       BASIC ''ASL  RN'' TEST WITH [DEST]=125252 AND C(0)
2439                                ;;*********************************************************************
2440  004646                        TST33:
2441  004646  012700  000033              MOV   #33,R0            ;;LOAD R0 WITH TEST NUMBER
2442  004652  012703  125252        1$:   MOV   #125252,R3        ;MAKE [DEST] = 125252
2443  004656  000257                      CCC                     ;MAKE C=0
2444
2445  004660  006303                2$:   ASL   R3                ;TEST THE ASL - IT SHOULD SET ''C''
2446
2447  004662  103402                      BCS   4$                ;BR IF ''C'' GOT SET
2448
2449  004664  000000                3$:   HALT                    ;ASL FAILED TO SET ''C'' BIT
2450  004666  000771                      BR    1$                ;LOCK ON HRD ERROR
2451
2452  004670  022703  052524        4$:   CMP   #52524,R3         ;WAS RESULT = 52524 ??
2453  004674  001402                      BEQ   TST34             ;;BR IF YES
2454
2455  004676  000000                5$:   HALT                    ;ASL DELIVERED THE WRONG RESULT
2456  004700  000764                      BR    1$                ;LOCK ON HARD ERROR
2457
2458                                ;;*********************************************************************
2459                                ;*TEST 34       BASIC ''ASL  RN'' TEST WITH [DEST]=052525 AND C(1)
2460                                ;;*********************************************************************
2461  004702                        TST34:
2462  004702  012700  000034              MOV   #34,R0            ;;LOAD R0 WITH TEST NUMBER
2463  004706  012703  052525        1$:   MOV   #052525,R3        ;MAKE [DEST] = 052525
2464  004712  000261                      SEC                     ;MAKE C=1
2465
2466  004714  006303                2$:   ASL   R3                ;TEST THE ASL - IT SHOULD CLR ''C''
2467
2468  004716  103002                      BCC   4$                ;BR IF ''C'' GOT CLEARED
2469
2470  004720  000000                3$:   HALT                    ;ASL FAILED TO CLEAR ''C''
2471  004722  000771                      BR    1$                ;LOCK ON HARD ERROR
2472
2473  004724  022703  125252        4$:   CMP   #125252,R3        ;RESULT = 125252 ??
2474  004730  001402                      BEQ   TST35             ;;BR IF YES
2475
2476  004732  000000                5$:   HALT                    ;ASL DELIVERED WRONG REULT
2477  004734  000764                      BR    1$                ;LOCK ON HARD ERROR
2478                                ;;*********************************************************************
2479                                ;*TEST 35       BASIC ''ROL  RN'' TEST WITH [DEST]=125252 AND C(0)
2480                                ;;*********************************************************************
2481  004736                        TST35:
2482  004736  012700  000035              MOV   #35,R0            ;;LOAD R0 WITH TEST NUMBER
2483  004742  012703  125252        1$:   MOV   #125252,R3        ;MAKE [DEST] = 125252
2484  004746  000257                      CCC                     ;MAKE C=0
2485
2486  004750  006103                2$:   ROL   R3                ;TEST THE ROL - IT SHOULD SET C
2487
2488  004752  103402                      BCS   4$                ;BR IF ''C'' GOT SET
2489
```

```
2490  004754  000000          3$:     HALT                    ;ROL FAILED TO SET ''C''
2491  004756  000771                  BR      1$              ;LOCK ON HARD ERROR
2492
2493  004760  022703  052524  4$:     CMP     #052524,R3      ;RESULT = 052524 ??
2494  004764  001402                  BEQ     TST36           ;;BR IF YES
2495
2496  004766  000000          5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2497  004770  000764                  BR      1$              ;LOCK ON HARD ERROR
2498
2499                          ;;**********************************************************************
2500                          ;*TEST 36       BASIC ''ROL  RN'' TEST WITH [DEST]=052524 AND C(1)
2501                          ;;**********************************************************************
2502  004772                  TST36:
2503  004772  012700  000036          MOV     #36,R0          ;;LOAD R0 WITH TEST NUMBER
2504  004776  012703  052524  1$:     MOV     #052524,R3      ;MAKE [DEST] = 052524
2505  005002  000261                  SEC                     ;MAKE C=1
2506
2507  005004  006103          2$:     ROL     R3              ;TEST THE ROL - IT SHOULD CLEAR C
2508
2509  005006  103002                  BCC     4$              ;BR IF ''C'' IS CLEAR
2510
2511  005010  000000          3$:     HALT                    ;ROL FAILED TO CLEAR ''C''
2512  005012  000771                  BR      1$              ;LOCK ON HARD ERROR
2513
2514  005014  022703  125251  4$:     CMP     #125251,R3      ;RESULT = 125251 ??
2515  005020  001402                  BEQ     TST37           ;;BR IF YES
2516
2517  005022  000000          5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2518  005024  000764                  BR      1$              ;LOCK ON HARD ERROR
2519
2520                          ;;**********************************************************************
2521                          ;*TEST 37       BASIS ''TSTB (R)'' TEST - EVEN ADDRESS
2522                          ;;**********************************************************************
2523  005026                  TST37:
2524  005026  012700  000037          MOV     #37,R0          ;;LOAD R0 WITH TEST NUMBER
2525  005032  012702  063330          MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2526  005036  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
2527  005042  000257          1$:     CCC                     ;SCOPE SYNC
2528
2529  005044  105712          2$:     TSTB    (R2)            ;TEST THE TSTB
2530
2531  005046  100402                  BMI     4$              ;BR IF 'N'' SET - IT SHOULD BE
2532
2533  005050  000000          3$:     HALT                    ;TSTB FAILED TO SET 'N''
2534  005052  000773                  BR      1$              ;LOCK ON HARD ERROR
2535
2536  005054  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2537  005056  001403                  BEQ     TST40           ;;BR IF NOT
2538
2539  005060  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2540  005062  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2541  005064  000766                  BR      1$              ;LOCK ON HARD EROR
2542                          ;;**********************************************************************
2543                          ;*TEST 40       BASIS ''TSTB (R)'' TEST - ODD ADDRESS
2544                          ;;**********************************************************************
2545  005066                  TST40:
```

```
 2546  005066  012700  000040          MOV     #40,R0          ;;LOAD R0 WITH TEST NUMBER
 2547  005072  012702  064040          MOV     #DWTB+6,R2      ;DEST ADDR = DWTB+6
 2548  005076  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
 2549  005102  012703  064041          MOV     #DWTB+7,R3      ;DEST ADDR USED = DWTB+7
 2550  005106  000257          1$:     CCC                     ;SCOPE SYNC
 2551
 2552  005110  105713          2$:     TSTB    (R3)            ;TEST THE TSTB
 2553
 2554  005112  100402                  BMI     4$              ;BR IF 'N' SET - IT SHOULD BE
 2555
 2556  005114  000000          3$:     HALT                    ;TSTB FAILED TO SET 'N'
 2557  005116  000773                  BR      1$              ;LOCK ON HARD ERROR
 2558
 2559  005120  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
 2560  005122  001403                  BEQ     TST41           ;;BR IF NOT
 2561
 2562  005124  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
 2563  005126  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
 2564  005130  000766                  BR      1$              ;LOCK ON HARD EROR
 2565
 2566                          ;;*******************************************************************
 2567                          ;*TEST 41        BASIC "TSTB @#A" TEST - EVEN ADDRESS
 2568                          ;;*******************************************************************
 2569  005132                  TST41:
 2570  005132  012700  000041          MOV     #41,R0          ;;LOAD R0 WITH TEST NUMBER
 2571  005136  012702  063326          MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
 2572  005142  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
 2573  005146  000257          1$:     CCC                     ;SCOPE SYNC
 2574
 2575  005150  105737  063326  2$:     TSTB    @#DWTA+4        ;TEST THE TSTB
 2576
 2577  005154  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
 2578
 2579  005156  000000          3$:     HALT                    ;TSTB FAILED TO SET "Z"
 2580  005160  000772                  BR      1$              ;LOCK ON HARD ERROR
 2581
 2582  005162  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
 2583  005164  001403                  BEQ     TST42           ;;BR IF NOT
 2584
 2585  005166  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
 2586  005170  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
 2587  005172  000765                  BR      1$              ;LOCK ON HARD ERROR
 2588
 2589                          ;;*******************************************************************
 2590                          ;*TEST 42        BASIC "TSTB @#A" TEST - ODD ADDRESS
 2591                          ;;*******************************************************************
 2592  005174                  TST42:
 2593  005174  012700  000042          MOV     #42,R0          ;;LOAD R0 WITH TEST NUMBER
 2594  005200  012702  063330          MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
 2595  005204  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 2596  005210  000257          1$:     CCC                     ;SCOPE SYNC
 2597
 2598  005212  105737  063331  2$:     TSTB    @#DWTA+7        ;TEST THE TSTB
 2599
 2600  005216  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
 2601
```

```
005220  000000                  3$:     HALT                    ;TSTB FAILED TO SET "Z"
005222  000772                          BR      1$              ;LOCK ON HARD ERROR

005224  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
005226  001403                          BEQ     TST43           ;;BR IF NOT

005230  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
005232  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
005234  000765                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*******************************************************************
                                ;*TEST 43        BASIC "DECB 1(SP)"
                                ;;*******************************************************************
005236                          TST43:
005236  012700  000043                  MOV     #43,R0          ;;LOAD R0 WITH TEST NUMBER
005242  010605                          MOV     SP,R5           ;SAVE SP
005244  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
005250  010506                  1$:     MOV     R5,SP
005252  005046                          CLR     -(SP)           ;[DEST] = 000000
005254  000257                          CCC                     ;SCOPE SYNC

005256  105366  000001          2$:     DECB    1(SP)           ;TEST THE DECB

005262  020416                          CMP     R4,(SP)         ;RESULT = 177400?
005264  001402                          BEQ     4$              ;BR IF YES

005266  000000                  3$:     HALT                    ;ERROR - DECB FAILED
005270  000767                          BR      1$              ;LOCK ON HARD ERROR

005272  010506                  4$:     MOV     R5,SP           ;RESET THE SP

                                ;;*******************************************************************
                                ;*TEST 44        BASIC "MOV @#A,R" TEST
                                ;;*******************************************************************
005274                          TST44:
005274  012700  000044                  MOV     #44,R0          ;;LOAD R0 WITH TEST NUMBER
005300  005003                  1$:     CLR     R3              ;[DEST] = 000000
005302  000257                          CCC                     ;SCOPE SYNC

005304  013703  063276          2$:     MOV     @#ATA,R3        ;TEST THE MOV

005310  022703  063322                  CMP     #DWTA,R3        ;RESULT = DWTA?
005314  001402                          BEQ     TST45           ;;BR IF YES

005316  000000                  3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
005320  000767                          BR      1$              ;LOCK ON HARD ERROR

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

```
2658  005344  000257                          CCC                     ;SCOPE SYNC
2659
2660  005346  012763  125252  000002  2$:     MOV     #125252,2(R3)   ;TEST THE MOV
2661
2662  005354  020412                          CMP     R4,(R2)         ;RESULT OK?
2663  005356  001402                          BEQ     TST46           ;;BR IF YES
2664
2665  005360  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2666  005362  000765                          BR      1$              ;LOCK ON HARD ERROR
2667
2668                                  ;;**********************************************************************
2669                                  ;*TEST 46        BASIC 'MOV #N,(R)'' TEST
2670                                  ;;**********************************************************************
2671  005364                          TST46:
2672  005364  012700  000046                 MOV     #46,R0          ;;LOAD R0 WITH TEST NUMBER
2673  005370  012703  063312                 MOV     #MBUF0,R3       ;DEST ADDR = MBUF0
2674  005374  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
2675  005400  005013                  1$:     CLR     (R3)            ;[DEST] = 000000
2676  005402  000257                          CCC                     ;SCOPE SYNC
2677
2678  005404  012713  125252          2$:     MOV     #125252,(R3)    ;TEST THE MOV
2679
2680  005410  020413                          CMP     R4,(R3)         ;RESULT OK?
2681  005412  001402                          BEQ     TST47           ;;BR IF YES
2682
2683  005414  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2684  005416  000770                          BR      1$              ;LOCK ON HARD ERROR
2685
2686                                  ;;**********************************************************************
2687                                  ;*TEST 47        BASIC 'MOV (RA)+,RB'' TEST
2688                                  ;;**********************************************************************
2689  005420                          TST47:
2690  005420  012700  000047                 MOV     #47,R0          ;;LOAD R0 WITH TEST NUMBER
2691  005424  012705  063276          1$:     MOV     #ATA,R5         ;SRC ADDR = ATA
2692  005430  005003                          CLR     R3              ;[DEST] = 000000
2693  005432  000257                          CCC                     ;SCOPE SYNC
2694
2695  005434  012503                  2$:     MOV     (R5)+,R3        ;TEST THE MOV
2696
2697  005436  022703  063322                 CMP     #DWTA,R3        ;RESULT OK?
2698  005442  000402                          BR      4$              ;BR IF YES
2699
2700  005444  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2701  005446  000766                          BR      1$              ;LOCK ON HARD ERROR
2702
2703  005450  022705  063300          4$:     CMP     #ATA+2,R5       ;DID SRC REG GET INCREMENTED?
2704  005454  001402                          BEQ     TST50           ;;BR IF YES
2705
2706  005456  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE SRC. REG.
2707  005460  000761                          BR      1$              ;LOCK ON HARD ERROR
2708
2709                                  ;;**********************************************************************
2710                                  ;*TEST 50        BASIC 'MOV  @#A,@#B''
2711                                  ;;**********************************************************************
2712  005462                          TST50:
2713  005462  012700  000050                 MOV     #50,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2714  005466  012702  063316          MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2715  005472  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2716  005476  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2717  005500  000257                          CCC                     ;SCOPE SYNC
2718
2719  005502  013737  063276  063316  2$:     MOV     @#ATA,@#MBUF1   ;TEST THE MOV
2720  005510  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ?
2721  005512  001402                          BEQ     TST51           ;;BR IF YES
2722
2723  005514  000000                  3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2724  005516  000767                          BR      1$              ;LOCK ON HARD ERROR
2725
2726                                  ;;*****************************************************************
2727                                  ;*TEST 51        BASIC 'MOV   X(R),PC'' TEST
2728                                  ;;*****************************************************************
2729  005520                          TST51:
2730  005520  012700  000051                  MOV     #51,R0          ;;LOAD R0 WITH TEST NUMBER
2731  005524  012705  005532          1$:     MOV     #2$,R5          ;[R5] = 2$ (BASE ADDRESS)
2732  005530  000257                          CCC                     ;SCOPE SYNC
2733
2734  005532  016507  000010          2$:     MOV     4$-2$(R5),PC    ;TEST THE MOV - GO TO NEXT TEST VIA 4$
2735
2736  005536  000000                  3$:     HALT                    ;MOV FAILED TO LOAD THE PC
2737  005540  000771                          BR      1$              ;LOCK ON HARD ERROR
2738
2739  005542  005544                  4$:     .+2                     ;POINTER TO NEXT TEST
2740
2741                                  ;;*****************************************************************
2742                                  ;*TEST 52        BASIC 'MOV  @#A,(R)'' TEST
2743                                  ;;*****************************************************************
2744  005544                          TST52:
2745  005544  012700  000052                  MOV     #52,R0          ;;LOAD R0 WITH TEST NUMBER
2746  005550  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2747  005554  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2748  005560  005012                  1$:     CLR     (R2)            ;MAKE [DEST]=000000
2749  005562  000257                          CCC                     ;SCOPE SYNC - Z=0
2750
2751  005564  013712  063276          2$:     MOV     @#ATA,(R2)      ;TEST THE MOV
2752
2753  005570  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ??
2754  005572  001402                          BEQ     TST53           ;;BR IF YES
2755
2756  005574  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2757  005576  000770                          BR      1$              ;LOCK ON HARD ERROR
2758
2759                                  ;;*****************************************************************
2760                                  ;*TEST 53        BASIC 'MOV  X(RA),RB'' TEST
2761                                  ;;*****************************************************************
2762  005600                          TST53:
2763  005600  012700  000053                  MOV     #53,R0          ;;LOAD R0 WITH TEST NUMBER
2764  005604  012705  063276                  MOV     #ATA,R5         ;[R5] = BASE ADDR FOR SOURCE (ATA)
2765  005610  005003                  1$:     CLR     R3              ;MAKE [DEST] = 000000
2766  005612  000257                          CCC                     ;SCOPE SYNC
2767
2768  005614  016503  000004          2$:     MOV     4(R5),R3        ;TEST THE MOV
2769
```

```
2770  005620  022703  064630          CMP     #DBTA,R3        ;RESULT = #DBTA ??
2771  005624  001402                  BEQ     TST54           ;;BR IF YES
2772
2773  005626  000000          3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2774  005630  000767                  BR      1$              ;LOCK ON HARD ERROR
2775
2776                          ;;**********************************************************************
2777                          ;*TEST 54        BASIC 'MOV  RA,-(RB)'' TEST
2778                          ;;**********************************************************************
2779  005632                  TST54:
2780  005632  012700  000054          MOV     #54,R0          ;;LOAD R0 WITH TEST NUMBER
2781  005636  012702  063312          MOV     #MBUF0,R2       ;FINAL DEST ADDR = MBUF0
2782  005642  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2783  005646  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = TEMP2 + 2
2784  005652  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2785  005654  000257                  CCC                     ;SCOPE SYNC
2786
2787  005656  010445          2$:     MOV     R4,-(R5)        ;TEST THE MOV
2788
2789  005660  020412                  CMP     R4,(R2)         ;RESULT = 125252
2790  005662  001402                  BEQ     4$              ;BR IF YES
2791
2792  005664  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG ESULT
2793  005666  000767                  BR      1$              ;LOCK ON HARD ERROR
2794
2795  005670  020205          4$:     CMP     R2,R5           ;DID REGISTER GET DECREMENTED ?
2796  005672  001402                  BEQ     TST55           ;;BR IF YES
2797
2798  005674  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2799  005676  000763                  BR      1$              ;LOCK ON HARD ERROR
2800
2801                          ;;**********************************************************************
2802                          ;*TEST 55        BASIC 'MOV   @#A,-(R)'' TEST
2803                          ;;**********************************************************************
2804  005700                  TST55:
2805  005700  012700  000055          MOV     #55,R0          ;;LOAD R0 WITH TEST NUMBER
2806  005704  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2807  005710  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2808  005714  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = MBUF0+2
2809  005720  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2810  005722  000257                  CCC                     ;SCOPE SYNC
2811
2812  005724  013745  063276  2$:     MOV     @#ATA,-(R5)     ;TEST THE MOV
2813
2814  005730  020412                  CMP     R4,(R2)         ;RESULT = 000000
2815  005732  001402                  BEQ     4$              ;BR IF YES
2816
2817  005734  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2818  005736  000766                  BR      1$              ;LOCK ON HARD ERROR
2819
2820  005740  020502          4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED ??
2821  005742  001402                  BEQ     TST56           ;;BR IF YES
2822
2823  005744  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2824  005746  000762                  BR      1$              ;LOCK ON HARD ERROR
2825
```

```
                                        ;;*****************************************************************
                                        ;*TEST 56        BASIC 'MOV  (R),@#A'' TEST
                                        ;;*****************************************************************
005750                                  TST56:
005750  012700  000056                          MOV     #56,R0          ;;LOAD R0 WITH TEST NUMBER
005754  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
005760  012704  063322                          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
005764  012705  063276                          MOV     #ATA,R5         ;SOURCE ADDR = ATA
005770  005012                          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
005772  000257                                  CCC                     ;SCOPE SYNC

005774  011537  063312                  2$:     MOV     (R5),@#MBUF0    ;TEST THE MOV

006000  020412                                  CMP     R4,(R2)         ;RESULT = #DWTA ??
006002  001402                                  BEQ     TST57           ;;BR IF YES

006004  000000                          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006006  000770                                  BR      1$              ;LOCK ON HARD ERROR

                                        ;;*****************************************************************
                                        ;*TEST 57        BASIC 'MOV  -(R),@#A'' TEST
                                        ;;*****************************************************************
006010                                  TST57:
006010  012700  000057                          MOV     #57,R0          ;;LOAD R0 WITH TEST NUMBER
006014  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006020  012704  063322                          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
006024  012705  063300                  1$:     MOV     #ATA+2,R5       ;INITIAL SOURCE ADDR = ATA+2
006030  005012                                  CLR     (R2)            ;MAKE [DEST] = 000000
006032  000257                                  CCC                     ;SCOPE SYNC

006034  014537  063312                  2$:     MOV     -(R5),@#MBUF0   ;TEST THE MOV

006040  020412                                  CMP     R4,(R2)         ;RESULT = #DWTA ?
006042  001402                                  BEQ     4$              ;BR IF YES

006044  000000                          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006046  000766                                  BR      1$              ;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   (RA),RB'' TEST
                                        ;;*****************************************************************
006062                                  TST60:
006062  012700  000060                          MOV     #60,R0          ;;LOAD R0 WITH TEST NUMBER
006066  012705  063276                  1$:     MOV     #ATA,R5         ;INITIAL SOURCE ADDR = ATA
006072  005003                                  CLR     R3              ;MAKE [DEST] = 000000
006074  000257                                  CCC                     ;SCOPE SYNC

006076  012503                          2$:     MOV     (R5)+,R3        ;TEST THE MOV

006100  022703  063322                          CMP     #DWTA,R3        ;RESULT = #DWTA ?
006104  001402                                  BEQ     4$              ;BR IF YES
```

```
2882
2883  006106  000000          3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2884  006110  000766                  BR      1$              ;LOCK ON HARD ERROR
2885
2886  006112  022705  063300  4$:     CMP     #ATA+2,R5       ;DID SOURCE REG GET INCREMENTED
2887  006116  001402                  BEQ     TST61           ;;BR IF YES
2888
2889  006120  000000          5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REGISTER
2890  006122  000761                  BR      1$              ;LOCK ON HARD ERROR
2891
2892                          ;;**********************************************************************
2893                          ;*TEST 61        BASIC 'MOV   X(RA),RB'' TEST
2894                          ;;**********************************************************************
2895  006124                  TST61:
2896  006124  012700  000061          MOV     #61,R0          ;;LOAD R0 WITH TEST NUMBER
2897  006130  012705  063276          MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
2898  006134  005003          1$:     CLR     R3              ;MAKE [DEST] = 000000
2899  006136  000257                  CCC                     ;SCOPE SYNC
2900
2901  006140  016503  000002  2$:     MOV     2(R5),R3        ;TEST THE MOV
2902
2903  006144  022703  064032          CMP     #DWTB,R3        ;RESULT = #DWTB ?
2904  006150  001402                  BEQ     TST62           ;;BR IF YES
2905
2906  006152  000000          3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2907  006154  000767                  BR      1$              ;LOCK ON HARD ERROR
2908                          ;;**********************************************************************
2909                          ;*TEST 62        BASIC 'MOV   @X(RA),RB'' TEST
2910                          ;;**********************************************************************
2911  006156                  TST62:
2912  006156  012700  000062          MOV     #62,R0          ;;LOAD R0 WITH TEST NUMBER
2913  006162  012737  063324  063314  MOV     #DWTA+2,@#MBUF0+2 ;SET UP ADDRESS TABLE MBUF0
2914  006170  012705  063312          MOV     #MBUF0,R5       ;BASE ADDRESS IN R5
2915  006174  005003          1$:     CLR     R3              ;MAKE [DEST] = 000000
2916  006176  000257                  CCC                     ;SCOPE SYNC
2917
2918  006200  017503  000002  2$:     MOV     @2(R5),R3       ;TEST THE MOV
2919
2920  006204  022703  177777          CMP     #-1,R3          ;RESULT = 177777
2921  006210  001402                  BEQ     TST63           ;;BR IF YES
2922
2923  006212  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2924  006214  000767                  BR      1$              ;LOCK ON HARD ERROR
2925
2926                          ;;**********************************************************************
2927                          ;*TEST 63        BASIC 'MOV  (R)+,X(R)'' TEST
2928                          ;;**********************************************************************
2929  006216                  TST63:
2930  006216  012700  000063          MOV     #63,R0          ;;LOAD R0 WITH TEST NUMBER
2931  006222  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2932  006226  012702  063320          MOV     #MBUF1+2,R2     ;FINAL DEST ADDR = MBUF1+2
2933  006232  010437  063312          MOV     R4,@#MBUF0      ;SOURCE OPERAND = 125252
2934  006236  012705  063312  1$:     MOV     #MBUF0,R5       ;[R5] = INITIAL SRC ADDR = MBUF0
2935  006242  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2936  006244  000257                  CCC                     ;SCOPE SYNC
2937
```

```
2938  006246  012565  000004          2$:     MOV     (R5)+,4(R5)     ;TEST THE MOV
2939
2940  006252  020412                          CMP     R4,(R2)         ;RESULT = 125252 ?
2941  006254  001402                          BEQ     4$              ;BR IF YES
2942
2943  006256  000000                  3$:     HALT                    ;MOV    DELIVERED WRONG RESULT
2944  006260  000766                          BR      1$              ;LOCK ON HARD ERROR
2945
2946  006262  022705  063314          4$:     CMP     #MBUF0+2,R5     ;DID REGISTER GET INCREMENTED ?
2947  006266  001402                          BEQ     TST64           ;;BR IF YES
2948
2949  006270  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2950  006272  000761                          BR      1$              ;LOCK ON HARD ERROR
2951
2952                                  ;;*****************************************************************
2953                                  ;*TEST 64        BASIC "CMP  R,@#A" TEST WITH [R] = [A]
2954                                  ;;*****************************************************************
2955  006274                          TST64:
2956  006274  012700  000064                  MOV     #64,R0          ;;LOAD R0 WITH TEST NUMBER
2957                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT1
2958  006300  032737  000002  063234          BIT     #BIT1,@#BPTLOC  ;BREAKPOINT HALT SET ??
2959  006306  001401                          BEQ     .+4             ;BR IF NOT
2960  006310  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2961  006312  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2962  006316  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
2963  006322  010405                  1$:     MOV     R4,R5           ;[R5] = SOURCE OP = 125252
2964  006324  010412                          MOV     R4,(R2)         ;MAKE [DEST] = 125252
2965  006326  000257                          CCC                     ;SCOPE SYNC
2966
2967  006330  020537  063312          2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2968
2969  006334  001402                          BEQ     4$              ;BR IF "Z" WAS SET - IT SHOULD BE
2970
2971  006336  000000                  3$:     HALT                    ;CMP FAILED TO SET "Z"
2972  006340  000770                          BR      1$              ;LOCK ON HARD ERROR
2973
2974  006342  020412                  4$:     CMP     R4,(R2)         ;IS RESULT STILL = 125252 ?
2975  006344  001402                          BEQ     TST65           ;;BR IF YES
2976
2977  006346  000000                  5$:     HALT                    ;CMP ALTERED [DEST]
2978  006350  000764                          BR      1$              ;LOCK ON HARD ERROR
2979
2980                                  ;;*****************************************************************
2981                                  ;*TEST 65        BASIC "CMP  R,@#A" WITH [R] NOT EQUAL TO [A]
2982                                  ;;*****************************************************************
2983  006352                          TST65:
2984  006352  012700  000065                  MOV     #65,R0          ;;LOAD R0 WITH TEST NUMBER
2985  006356  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2986  006362  012704  125252                  MOV     #125252,R4      ;MAKE RESULT S / B = 125252
2987  006366  005005                  1$:     CLR     R5              ;[R5] = SOURCE OP = 000000
2988  006370  010412                          MOV     R4,(R2)         ;MAKE [DEST] = 125252
2989  006372  000277                          SCC                     ;SCOPE SYNC - MAKE Z=1
2990
2991  006374  020537  063312          2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2992
2993  006400  001002                          BNE     TST66           ;.BR IF Z=0 - IT SHOULD BE
```

```
2994
2995  006402  000000                  3$:     HALT                    ;CMP FAILED TO CLEAR ''Z''
2996  006404  000770                          BR      1$              ;LOCK ON HARD ERROR
2997
2998                                  ;;**********************************************************************
2999                                  ;*TEST 66        BASIC 'BIS #N,@#A'' TEST - N=177777,[A]=000000
3000                                  ;;**********************************************************************
3001  006406                          TST66:
3002  006406  012700  000066                 MOV     #66,R0          ;;LOAD R0 WITH TEST NUMBER
3003  006412  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3004  006416  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3005  006422  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3006  006424  000257                          CCC                     ;SCOPE SYNC
3007
3008  006426  052737  177777  063312  2$:     BIS     #-1,@#MBUF0     ;TEST THE BIS
3009
3010  006434  020412                          CMP     R4,(R2)         ;RESULT OK?
3011  006436  001402                          BEQ     TST67           ;;BR IF YES
3012
3013  006440  000000                  3$:     HALT                    ;BIS FAILED TO SET ALL BITS IN BITFLG
3014  006442  000767                          BR      1$              ;LOCK ON HARD ERROR
3015
3016                                  ;;**********************************************************************
3017                                  ;*TEST 67        BASIC 'BIC  #N,@#A'' TEST
3018                                  ;;**********************************************************************
3019  006444                          TST67:
3020  006444  012700  000067                 MOV     #67,R0          ;;LOAD R0 WITH TEST NUMBER
3021  006450  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3022  006454  012704  000077                 MOV     #77,R4          ;RESULT S / B = 77
3023  006460  012712  177777          1$:     MOV     #-1,(R2)        ;MAKE [DEST] = 177777
3024  006464  000257                          CCC                     ;SCOPE SYNC
3025
3026  006466  042737  177700  063312  2$:     BIC     #177700,@#MBUF0 ;TEST THE BIC
3027
3028  006474  020412                          CMP     R4,(R2)         ;DID RESULT = 77 ?  .
3029  006476  001402                          BEQ     TST70           ;;BR IF YES
3030
3031  006500  000000                  3$:     HALT                    ;BIC DELIVERED THE WRONG RESULT
3032  006502  000766                          BR      1$              ;LOCK ON HARD ERROR
3033
3034                                  ;;**********************************************************************
3035                                  ;*TEST 70        BASIC 'BIC #N,R'' TEST
3036                                  ;;**********************************************************************
3037  006504                          TST70:
3038  006504  012700  000070                 MOV     #70,R0          ;;LOAD R0 WITH TEST NUMBER
3039  006510  005003                  1$:     CLR     R3              ;[DEST] = 177777
3040  006512  005103                          COM     R3
3041  006514  000257                          CCC                     ;SCOPE SYNC
3042
3043  006516  042703  177400          2$:     BIC     #177400,R3      ;TEST THE BIC
3044
3045  006522  022703  000377                 CMP     #377,R3         ;RESULT OK?
3046  006526  001402                          BEQ     TST71           ;;BR IF YES
3047
3048  006530  000000                  3$:     HALT                    ;BIC FAILED TO CLEAR HI-BYTE
3049  006532  000766                          BR      1$              ;LOCK ON HARD ERROR
```

```
                                        ;;*******************************************************************
                                        ;*TEST 71        BASIC ''BIC #N,2(SP)'' TEST
                                        ;;*******************************************************************
006534                                  TST71:
006534  012700  000071                          MOV     #71,R0          ;;LOAD R0 WITH TEST NUMBER
006540  012704  000357                          MOV     #357,R4         ;RESULT S / B = 357
006544  010605                                  MOV     SP,R5           ;SAVE SP
006546  010506                          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
006550  012746  000377                          MOV     #377,-(SP)      ;[DEST] = 377 PUT ON STACK
006554  005746                                  TST     -(SP)           ;DECREMENT SP
006556  000257                                  CCC                     ;SCOPE SYNC

006560  042766  000020  000002          2$:     BIC     #20,2(SP)       ;TEST THE BIC - CLEAR BIT 4

006566  010602                                  MOV     SP,R2           ;[R2] = DEST ADDR
006570  005722                                  TST     (R2)+
006572  020412                                  CMP     R4,(R2)         ;RESULT = 357?
006574  001402                                  BEQ     4$              ;BR IF YES

006576  000000                          3$:     HALT                    ;BIC FAILED TO CLR BIT2 OF DEST
006600  000762                                  BR      1$              ;LOCK ON HARD ERROR

006602  010506                          4$:     MOV     R5,SP

                                        ;;*******************************************************************
                                        ;*TEST 72        BASIC ''ADD  #N,RN '' TEST
                                        ;;*******************************************************************
006604                                  TST72:
006604  012700  000072                          MOV     #72,R0          ;;LOAD R0 WITH TEST NUMBER
006610  012703  000002                  1$:     MOV     #2,R3           ;MAKE [DEST] = 2
006614  000257                                  CCC                     ;SCOPE SYNC

006616  062703  000002                  2$:     ADD     #2,R3           ;TEST THE ADD

006622  022703  000004                          CMP     #4,R3           ;RESULT = 4 ?
006626  001402                                  BEQ     TST73           ;;BR IF YES

006630  000000                          3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
006632  000766                                  BR      1$              ;LOCK ON HARD ERROR

                                        ;;*******************************************************************
                                        ;*TEST 73        BASIC ''ADD  #N,(R) '' TEST
                                        ;;*******************************************************************
006634                                  TST73:
006634  012700  000073                          MOV     #73,R0          ;;LOAD R0 WITH TEST NUMBER
006640  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006644  012704  000004                          MOV     #4,R4           ;RESULT S / B = 4
006650  012712  000002                  1$:     MOV     #2,(R2)         ;MAKE [DEST] = 2
006654  000257                                  CCC                     ;SCOPE SYNC

006656  062712  000002                  2$:     ADD     #2,(R2) ;TEST THE ADD

006662  020412                                  CMP     R4,(R2)         ;RESULT = 4 ?
006664  001402                                  BEQ     TST74           ;;BR IF YES
```

```
3106  006666  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3107  006670  000767                          BR      1$              ;LOCK ON HARD ERROR
3108
3109                                  ;;**********************************************************************
3110                                  ;*TEST 74        BASIC ''ADD  #N,X(R)'' TEST
3111                                  ;;**********************************************************************
3112  006672                          TST74:
3113  006672  012700  000074                MOV     #74,R0          ;;LOAD R0 WITH TEST NUMBER
3114  006676  012704  000002                MOV     #2,R4           ;RESULT S / B = 2
3115  006702  012702  063314                MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0 + 2
3116  006706  012705  063312          1$:   MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
3117  006712  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
3118  006714  000257                        CCC                     ;SCOPE SYNC
3119
3120  006716  062765  000002  000002  2$:   ADD     #2,2(R5)        ;TEST THE ADD
3121
3122  006724  020412                        CMP     R4,(R2)         ;RESULT = 2 ?
3123  006726  001402                        BEQ     TST75           ;;BR IF YES
3124
3125  006730  000000                  3$:   HALT                    ;ADD DELIVERED THE WRONG RESULT
3126  006732  000765                        BR      1$              ;LOOP ON HARD ERROR
3127
3128                                  ;;**********************************************************************
3129                                  ;*TEST 75        BASIC ''CMPB #N,(SP)+'' TEST
3130                                  ;;**********************************************************************
3131  006734                          TST75:
3132  006734  012700  000075                MOV     #75,R0          ;;LOAD R0 WITH TEST NUMBER
3133  006740  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3134  006744  010605                        MOV     SP,R5           ;SAVE SP
3135  006746  010602                        MOV     SP,R2           ;SET UP DEST ADDR
3136  006750  005742                        TST     -(R2)           ;R2 CONTAINS DEST ADDR
3137  006752  010506                  1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3138  006754  010446                        MOV     R4,-(SP)        ;MAKE [DEST] = 177400
3139  006756  000257                        CCC                     ;SCOPE SYNC - ''Z'' = 0
3140
3141  006760  122726  000000          2$:   CMPB    #0,(SP)+        ;TEST THE CMPB
3142
3143  006764  001402                        BEQ     4$              ;BR IF ''Z'' SET - IT SHOULD BE
3144
3145  006766  000000                  3$:   HALT                    ;CMPB FAILED TO SET ''Z''
3146  006770  000770                        BR      1$              ;LOCK ON HARD ERROR
3147
3148  006772  020506                  4$:   CMP     R5,SP           ;DID SP GET UPDATED BY 2?
3149  006774  001402                        BEQ     6$              ;BR IF YES
3150
3151  006776  000000                  5$:   HALT                    ;CMPB FAILED TO UPDATE SP PROPERLY
3152  007000  000764                        BR      1$              ;LOCK ON HARD ERROR
3153
3154  007002  020412                  6$:   CMP     R4,(R2)         ;[DEST] ALTERED?
3155  007004  001402                        BEQ     TST76           ;;BR IF NOT
3156
3157  007006  000000                  7$:   HALT                    ;CMPB MODIFIED [DEST]
3158  007010  000760                        BR      1$              ;LOCK ON HARD ERROR.
3159
3160                                  ;;**********************************************************************
3161                                  ;*TEST 76        BASIC ''CMPB (RA)+,(RB)+'' - SRC AND DEST EVEN
```

```
                        ;;*****************************************************************
007012                  TST76:
007012  012700  000076          MOV     #76,R0          ;;LOAD R0 WITH TEST NUMBER
007016  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
007022  012702  063324          MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
007026  012705  063330  1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
007032  010203                  MOV     R2,R3           ;R3 GETS DEST ADDR
007034  000257                  CCC                     ;SCOPE SYNC
                        
007036  122523          2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
                        
007040  001402                  BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
                        
007042  000000          3$:     HALT                    ;CMPB FAILED TO SET ''Z''
007044  000770                  BR      1$              ;LOCK ON HARD ERROR
                        
007046  022703  063325  4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
007052  001402                  BEQ     6$              ;BR IF YES
                        
007054  000000          5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
007056  000763                  BR      1$              ;LOCK ON HARD ERROR
                        
007060  022705  063331  6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
007064  001402                  BEQ     8$              ;BR IF YES
                        
007066  000000          7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
007070  000756                  BR      1$              ;LOCK ON HARD ERROR
                        
007072  020412          8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
007074  001403                  BEQ     TST77           ;;BR IF NOT
                        
007076  000000          9$:     HALT                    ;CMPB DELIVERED A RESULT
007100  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
007102  000751                  BR      1$              ;LOCK ON HARD ERROR
                        
                        ;;*****************************************************************
                        ;*TEST 77        BASIC ''CMPB (RA)+,(RB)+'' - SRC AND DEST ODD
                        ;;*****************************************************************
007104                  TST77:
007104  012700  000077          MOV     #77,R0          ;;LOAD R0 WITH TEST NUMBER
007110  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
007114  012702  063324          MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
007120  012705  063327  1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
007124  012703  063325          MOV     #DWTA+3,R3      ;R3 GETS DEST ADDR+1
007130  000257                  CCC                     ;SCOPE SYNC
                        
007132  122523          2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
                        
007134  001402                  BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE
                        
007136  000000          3$:     HALT                    ;CMPB FAILED TO SET ''Z''
007140  000767                  BR      1$              ;LOCK ON HARD ERROR
                        
007142  022703  063326  4$:     CMP     #DWTA+4,R3      ;DID DEST REG GET UPDATED?
007146  001402                  BEQ     6$              ;BR IF YES
```

```
3218  007150  000000          5$:   HALT                  ;CMPB FAILED TO UPDATE DEST REG
3219  007152  000762                BR    1$              ;LOCK ON HARD ERROR
3220
3221  007154  022705  063330  6$:   CMP   #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3222  007160  001402                BEQ   8$              ;BR IF YES
3223
3224  007162  000000          7$:   HALT                  ;CMPB FAILED TO UPDATE SRC REG
3225  007164  000755                BR    1$              ;LOCK ON HARD ERROR
3226
3227  007166  020412          8$:   CMP   R4,(R2)         ;DID [DEST] GET ALTERED?
3228  007170  001403                BEQ   TST100          ;;BR IF NOT
3229
3230  007172  000000          9$:   HALT                  ;CMPB DELIVERED A RESULT
3231  007174  010412                MOV   R4,(R2)         ;RESTORE [DEST]
3232  007176  000750                BR    1$              ;LOCK ON HARD ERROR
3233
3234                        ;;*****************************************************************
3235                        ;*TEST 100      BASIC ''CMPB (RA)+,(RB)+'' - SRC / EVEN,DEST / ODD
3236                        ;;*****************************************************************
3237  007200                TST100:
3238  007200  012700  000100       MOV   #100,R0         ;;LOAD R0 WITH TEST NUMBER
3239  007204  012704  177400       MOV   #177400,R4      ;RESULT S / B = 177400
3240  007210  012702  063326       MOV   #DWTA+4,R2      ;DEST ADDR = DWTA+4
3241  007214  012705  063330 1$:   MOV   #DWTA+6,R5      ;SRC ADDR = DWTA+6
3242  007220  012703  063327       MOV   #DWTA+5,R3      ;R3 GETS DEST ADDR
3243  007224  000257                CCC                   ;SCOPE SYNC
3244
3245  007226  122523          2$:   CMPB  (R5)+,(R3)+     ;TEST THE CMPB
3246
3247  007230  001402                BEQ   4$              ;BR IF 'Z'' = 1 - IT SHOULD BE
3248
3249  007232  000000          3$:   HALT                  ;CMPB FAILED TO SET 'Z''
3250  007234  000767                BR    1$              ;LOCK ON HARD ERROR
3251
3252  007236  022703  063330  4$:   CMP   #DWTA+6,R3      ;DID DEST REG GET UPDATED?
3253  007242  001402                BEQ   6$              ;BR IF YES
3254
3255  007244  000000          5$:   HALT                  ;CMPB FAILED TO UPDATE DEST REG
3256  007246  000762                BR    1$              ;LOCK ON HARD ERROR
3257
3258  007250  022705  063331  6$:   CMP   #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3259  007254  001402                BEQ   8$              ;BR IF YES
3260
3261  007256  000000          7$:   HALT                  ;CMPB FAILED TO UPDATE SRC REG
3262  007260  000755                BR    1$              ;LOCK ON HARD ERROR
3263
3264  007262  020412          8$:   CMP   R4,(R2)         ;DID [DEST] GET ALTERED?
3265  007264  001403                BEQ   TST101          ;;BR IF NOT
3266
3267  007266  000000          9$:   HALT                  ;CMPB DELIVERED A RESULT
3268  007270  010412                MOV   R4,(R2)         ;RESTORE [DEST]
3269  007272  000750                BR    1$              ;LOCK ON HARD ERROR
3270
3271                        ;;*****************************************************************
3272                        ;*TEST 101      BASIC ''CMPB (RA)+,(RB)+'' - SRC / ODD,DEST / EVEN
3273                        ;;*****************************************************************
```

```
007274                          TST101:
007274  012700  000101                  MOV     #101,R0         ;;LOAD R0 WITH TEST NUMBER
007300  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
007304  012702  063324                  MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
007310  012705  063327          1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
007314  010203                          MOV     R2,R3           ;R3 GETS DEST ADDR
007316  000257                          CCC                     ;SCOPE SYNC

007320  122523                  2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB

007322  001402                          BEQ     4$              ;BR IF ''Z'' = 1 - IT SHOULD BE

007324  000000                  3$:     HALT                    ;CMPB FAILED TO SET ''Z''
007326  000770                          BR      1$              ;LOCK ON HARD ERROR

007330  022703  063325          4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
007334  001402                          BEQ     6$              ;BR IF YES

007336  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
007340  000763                          BR      1$              ;LOCK ON HARD ERROR

007342  022705  063330          6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
007346  001402                          BEQ     8$              ;BR IF YES

007350  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
007352  000756                          BR      1$              ;LOCK ON HARD ERROR

007354  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
007356  001403                          BEQ     TST102          ;;BR IF NOT

007360  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
007362  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
007364  000751                          BR      1$              ;LOCK ON HARD ERROR


                                ;;*******************************************************************
                                ;*TEST 102       BASIC 'MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN
                                ;;*******************************************************************
007366                          TST102:
007366  012700  000102                  MOV     #102,R0         ;;LOAD R0 WITH TEST NUMBER
007372  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
007376  012703  063312                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
007402  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
007406  012705  064630          1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
007412  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
007416  000257                          CCC                     ;SCOPE SYNC

007420  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB

007424  020412                          CMP     R4,(R2)         ;RESULT OK?
007426  001402                          BEQ     4$              ;BR IF YES

007430  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
007432  000765                          BR      1$              ;LOCK ON HARD ERROR

007434  022705  064631          4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
007440  001402                          BEQ     TST103          ;;BR IF YES
```

```
                                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
007442  000000                        BR      1$              ;LOCK ON HARD ERROR
007444  000760                ;;**********************************************************************
                                ;*TEST 103       BASIC 'MOVB (RA)+,X(RB) - SRC ODD / DEST ODD
                                ;;**********************************************************************
007446                          TST103:
007446  012700  000103                MOV     #103,R0         ;;LOAD R0 WITH TEST NUMBER
007452  012702  063316                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
007456  012703  063312                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
007462  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
007466  012705  064635        1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
007472  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
007476  000257                        CCC                     ;SCOPE SYNC

007500  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB

007504  020412                        CMP     R4,(R2)         ;RESULT OK?
007506  001402                        BEQ     4$              ;BR IF YES

007510  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
007512  000765                        BR      1$              ;LOCK ON HARD ERROR

007514  022705  064636        4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
007520  001402                        BEQ     TST104          ;;BR IF YES

007522  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
007524  000760                        BR      1$              ;LOCK ON HARD ERROR
                                ;;**********************************************************************
                                ;*TEST 104       BASIC 'MOVB (RA)+,X(RB) - SRC EVEN / DEST ODD
                                ;;**********************************************************************
007526                          TST104:
007526  012700  000104                MOV     #104,R0         ;;LOAD R0 WITH TEST NUMBER
007532  012702  063316                MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
007536  012703  063312                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
007542  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
007546  012705  064630        1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
007552  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
007556  000257                        CCC                     ;SCOPE SYNC

007560  112563  000005        2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB

007564  020412                        CMP     R4,(R2)         ;RESULT OK?
007566  001402                        BEQ     4$              ;BR IF YES

007570  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
007572  000765                        BR      1$              ;LOCK ON HARD ERROR

007574  022705  064631        4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
007600  001402                        BEQ     TST105          ;;BR IF YES

007602  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
007604  000760                        BR      1$              ;LOCK ON HARD ERROR
                                ;;**********************************************************************
                                ;*TEST 105       BASIC 'MOVB (RA)+,X(RB) - SRC ODD / DEST EVEN
                                ;;**********************************************************************
```

```
3386  007606
3387  007606  012700  000105          TST105:
                                              MOV     #105,R0         ;;LOAD R0 WITH TEST NUMBER
3388  007612  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3389  007616  012703  063312                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3390  007622  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3391  007626  012705  064635          1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3392  007632  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
3393  007636  000257                          CCC                     ;SCOPE SYNC
3394
3395  007640  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3396
3397  007644  020412                          CMP     R4,(R2)         ;RESULT OK?
3398  007646  001402                          BEQ     4$              ;BR IF YES
3399
3400  007650  000000          3$:     HALT                            ;MOV DELIVERED WRONG RESULT
3401  007652  000765                          BR      1$              ;LOCK ON HARD ERROR
3402
3403  007654  022705  064636          4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3404  007660  001402                          BEQ     TST106          ;;BR IF YES
3405
3406  007662  000000          5$:     HALT                            ;MOVB FAILED TO UPDATE SRC REG
3407  007664  000760                          BR      1$              ;LOCK ON HARD ERROR
3408
3409                          ;;**********************************************************************
3410                          ;*TEST 106      BASIC 'MOVB 2(RA),(RB)+'' TEST - SRC EVEN / DEST EVEN
3411                          ;;**********************************************************************
3412  007666
3413  007666  012700  000106          TST106:
                                              MOV     #106,R0         ;;LOAD R0 WITH TEST NUMBER
3414  007672  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3415  007676  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3416  007702  012705  064032                  MOV     #DWTB,R5        ;SRC ADDR = DWTB
3417  007706  010203          1$:     MOV     R2,R3                   ;R3 GETS DEST ADDR
3418  007710  012713  177777                  MOV     #-1,(R3)        ;[DEST] = 177400
3419  007714  000257                          CCC                     ;SCOPE SYNC
3420
3421  007716  116523  000002          2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3422
3423  007722  020412                          CMP     R4,(R2)         ;RESULT OK?
3424  007724  001402                          BEQ     4$              ;BR IF YES
3425
3426  007726  000000          3$:     HALT                            ;MOVB DELIVERED WRONG RESULT
3427  007730  000766                          BR      1$              ;LOCK ON HARD ERROR
3428
3429  007732  022703  063313          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3430  007736  001402                          BEQ     TST107          ;;BR IF YES
3431
3432  007740  000000          5$:     HALT                            ;MOVB FAILED TO AUTO INCREMENT DEST REG
3433  007742  000761                          BR      1$              ;LOCK ON HARD ERROR
3434
3435                          ;;**********************************************************************
3436                          ;*TEST 107      BASIC 'MOVB 2(RA),(RB)+'' TEST - SRC ODD / DEST EVEN
3437                          ;;**********************************************************************
3438  007744
3439  007744  012700  000107          TST107:
                                              MOV     #107,R0         ;;LOAD R0 WITH TEST NUMBER
3440  007750  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3441  007754  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
```

```
3442  007760  012705  064634          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3443  007764  010203          1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3444  007766  012713  177777          MOV     #-1,(R3)        ;[DEST] = 177777
3445  007772  000257                  CCC                     ;SCOPE SYNC
3446
3447  007774  116523  000001  2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3448
3449  010000  020412                  CMP     R4,(R2)         ;RESULT OK?
3450  010002  001402                  BEQ     4$              ;BR IF YES
3451
3452  010004  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3453  010006  000766                  BR      1$              ;LOCK ON HARD ERROR
3454
3455  010010  022703  063313  4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3456  010014  001402                  BEQ     TST110          ;;BR IF YES
3457
3458  010016  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3459  010020  000761                  BR      1$              ;LOCK ON HARD ERROR
3460
3461                          ;;*****************************************************************
3462                          ;*TEST 110      BASIC 'MOVB 2(RA),(RB)+' TEST - SRC EVEN / DEST ODD
3463                          ;;*****************************************************************
3464  010022                  TST110:
3465  010022  012700  000110          MOV     #110,R0         ;;LOAD R0 WITH TEST NUMBER
3466  010026  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3467  010032  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3468  010036  012705  064032          MOV     #DWTB,R5        ;SRC ADDR = DWTB
3469  010042  012703  063313  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR
3470  010046  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3471  010052  000257                  CCC                     ;SCOPE SYNC
3472
3473  010054  116523  000002  2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3474
3475  010060  020412                  CMP     R4,(R2)         ;RESULT OK?
3476  010062  001402                  BEQ     4$              ;BR IF YES
3477
3478  010064  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3479  010066  000765                  BR      1$              ;LOCK ON HARD ERROR
3480
3481  010070  022703  063314  4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3482  010074  001402                  BEQ     TST111          ;;BR IF YES
3483
3484  010076  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3485  010100  000760                  BR      1$              ;LOCK ON HARD ERROR
3486
3487                          ;;*****************************************************************
3488                          ;*TEST 111      BASIC 'MOVB 2(RA),(RB)+' TEST - SRC ODD / DEST ODD
3489                          ;;*****************************************************************
3490  010102                  TST111:
3491  010102  012700  000111          MOV     #111,R0         ;;LOAD R0 WITH TEST NUMBER
3492  010106  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3493  010112  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3494  010116  012705  064634          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3495  010122  012703  063313  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR = MBUF0+1
3496  010126  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3497  010132  000257                  CCC                     ;SCOPE SYNC
```

```
3498
3499  010134  116523  000001        2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3500
3501  010140  020412                        CMP     R4,(R2)         ;RESULT OK?
3502  010142  001402                        BEQ     4$              ;BR IF YES
3503
3504  010144  000000                3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3505  010146  000765                        BR      1$              ;LOCK ON HARD ERROR
3506
3507  010150  022703  063314        4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3508  010154  001402                        BEQ     TST112          ;;BR IF YES
3509
3510  010156  000000                5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3511  010160  000760                        BR      1$              ;LOCK ON HARD ERROR
3512
3513                                ;;**********************************************************************
3514                                ;*TEST 112       BASIC 'MOVB -(RA),RB'' TEST - SRC EVEN ADDR
3515                                ;;**********************************************************************
3516  010162                        TST112:
3517  010162  012700  000112                MOV     #112,R0         ;;LOAD R0 WITH TEST NUMBER
3518  010166  012705  063331        1$:     MOV     #DWTA+7,R5      ;SRC ADDR = DWTA+7
3519  010172  005003                        CLR     R3              ;[DEST] = 000000
3520  010174  000257                        CCC                     ;SCOPE SYNC
3521
3522  010176  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3523
3524  010200  022703  177777                CMP     #-1,R3          ;RESULT OK?
3525  010204  001402                        BEQ     4$              ;BR IF YES
3526
3527  010206  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3528  010210  000766                        BR      1$              ;LOCK ON HARD ERROR
3529
3530  010212  022705  063330        4$:     CMP     #DWTA+6,R5      ;SRC REG GET DECREMENTED?
3531  010216  001402                        BEQ     TST113          ;;BR IF YES
3532
3533  010220  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3534  010222  000761                        BR      1$              ;LOCK ON HARD ERROR
3535                                ;;**********************************************************************
3536                                ;*TEST 113       BASIC 'MOVB -(RA),RB'' TEST - SRC ODD ADDR
3537                                ;;**********************************************************************
3538  010224                        TST113:
3539  010224  012700  000113                MOV     #113,R0         ;;LOAD R0 WITH TEST NUMBER
3540  010230  012705  063330        1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3541  010234  005003                        CLR     R3              ;[DEST] = 000000
3542  010236  000257                        CCC                     ;SCOPE SYNC
3543
3544  010240  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3545
3546  010242  022703  177777                CMP     #-1,R3          ;RESULT OK?
3547  010246  001402                        BEQ     4$              ;BR IF YES
3548
3549  010250  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3550  010252  000766                        BR      1$              ;LOCK ON HARD ERROR
3551
3552  010254  022705  063327        4$:     CMP     #DWTA+5,R5      ;SRC REG GET DECREMENTED?
3553  010260  001402                        BEQ     TST114          ;;BR IF YES
```

```
3554
3555  010262  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3556  010264  000761                        BR      1$              ;LOCK ON HARD ERROR
3557
3558                                ;;**********************************************************************
3559                                ;*TEST 114       BASIC 'MOVB (RA)+,-(SP)'' TEST - SRC ADDR EVEN
3560                                ;;**********************************************************************
3561  010266                        TST114:
3562  010266  012700  000114                MOV     #114,R0         ;;LOAD R0 WITH TEST NUMBER
3563  010272  010605                        MOV     SP,R5           ;SAVE SP
3564  010274  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3565  010300  010506                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3566  010302  012703  064630                MOV     #DBTA,R3        ;SRC ADDR = DBTA
3567  010306  012746  177777                MOV     #-1,-(SP)       ;[DEST] = 177777
3568  010312  010602                        MOV     SP,R2           ;R2 GETS DEST ADDR
3569  010314  005726                        TST     (SP)+           ;RESET SP
3570  010316  000257                        CCC                     ;SCOPE SYNC
3571
3572  010320  112346                2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3573
3574  010322  022703  064631                CMP     #DBTA+1,R3      ;DID MOVB INCREMENT SRC REG?
3575  010326  001402                        BEQ     4$              ;BR IF YES
3576
3577  010330  000000                3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3578  010332  000762                        BR      1$              ;LOCK ON HARD ERROR
3579
3580  010334  020412                4$:     CMP     R4,(R2)         ;RESULT OK?
3581  010336  001402                        BEQ     6$              ;BR IF YES
3582
3583  010340  000000                        HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3584  010342  000756                        BR      1$              ;LOCK ON HARD ERROR
3585
3586  010344  020206                6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2 ?
3587  010346  001402                        BEQ     8$              ;BR IF YES
3588
3589  010350  000000                        HALT                    ;MOVB FAILED TO PUSH SP PROPERLY
3590  010352  000752                        BR      1$              ;LOCK ON HARD ERROR
3591
3592  010354  010506                8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3593
3594                                ;;**********************************************************************
3595                                ;*TEST 115       BASIC 'MOVB (RA)+,-(SP)'' TEST - SRC ADDR ODD
3596                                ;;**********************************************************************
3597  010356                        TST115:
3598  010356  012700  000115                MOV     #115,R0         ;;LOAD R0 WITH TEST NUMBER
3599  010362  010605                        MOV     SP,R5           ;SAVE SP
3600  010364  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3601  010370  010506                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3602  010372  012703  064035                MOV     #DWTB+3,R3      ;SRC ADDR = DWTB+3
3603  010376  012746  177777                MOV     #-1,-(SP)       ;[DEST] = 177777
3604  010402  010602                        MOV     SP,R2           ;R2 GETS DEST ADDR
3605  010404  005726                        TST     (SP)+           ;RESET SP
3606  010406  000257                        CCC                     ;SCOPE SYNC
3607
3608  010410  112346                2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3609
```

```
3610  010412  022703  064036          CMP     #DWTB+4,R3      ;DID MOVB INCREMENT SRC REG?
3611  010416  001402                  BEQ     4$              ;BR IF YES
3612
3613  010420  000000          3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3614  010422  000762                  BR      1$              ;LOCK ON HARD ERROR
3615
3616  010424  020412          4$:     CMP     R4,(R2)         ;RESULT OK?
3617  010426  001402                  BEQ     6$              ;BR IF YES
3618
3619  010430  000000                  HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3620  010432  000756                  BR      1$              ;LOCK ON HARD ERROR
3621
3622  010434  020206          6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2
3623  010436  001402                  BEQ     8$              ;BR IF YES
3624
3625  010440  000000                  HALT                    ;MOVB FAILED TO PUSH SP
3626  010442  000752                  BR      1$              ;LOCK ON HARD ERROR
3627
3628  010444  010506          8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3629
3630                          ;;*********************************************************************
3631                          ;*TEST 116      BASIC 'MOVB X(R),@#A'' TEST - SRC EVEN / DEST EVEN
3632                          ;;*********************************************************************
3633  010446                  TST116:
3634  010446  012700  000116          MOV     #116,R0         ;;LOAD R0 WITH TEST NUMBER
3635  010452  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3636  010456  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
3637  010462  012705  064032          MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3638  010466  005012          1$:     CLR     (R2)            ;[DEST] = 000000
3639  010470  000257                  CCC                     ;SCOPE SYNC
3640
```

```
3641  010472  116537  000006  063312  2$:     MOVB    6(R5),@#MBUF0   ;TEST THE MOVB
3642
3643  010500  020412                          CMP     R4,(R2)         ;RESULT OK?
3644  010502  001402                          BEQ     TST117          ;;BR IF YES
3645
3646  010504  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3647  010506  000767                          BR      1$              ;LOCK ON HARD ERRROR
3648                                  ;;**********************************************************************
3649                                  ;*TEST 117       BASIC 'MOVB X(R),@#A'' TEST - SRC ODD / DEST EVEN
3650                                  ;;**********************************************************************
3651  010510                          TST117:
3652  010510  012700  000117                  MOV     #117,R0         ;;LOAD R0 WITH TEST NUMBER
3653  010514  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3654  010520  012704  000001                  MOV     #1,R4           ;RESULT S / B = 1
3655  010524  012705  064634                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3656  010530  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3657  010532  000257                          CCC                     ;SCOPE SYNC
3658
3659  010534  116537  000001  063312  2$:     MOVB    1(R5),@#MBUF0   ;TEST THE MOVB
3660
3661  010542  020412                          CMP     R4,(R2)         ;RESULT OK?
3662  010544  001402                          BEQ     TST120          ;;BR IF YES
3663
3664  010546  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3665  010550  000767                          BR      1$              ;LOCK ON HARD ERRROR
3666                                  ;;**********************************************************************
3667                                  ;*TEST 120       BASIC 'MOVB X(R),@#A'' TEST - SRC EVEN / DEST ODD
3668                                  ;;**********************************************************************
3669  010552                          TST120:
3670  010552  012700  000120                  MOV     #120,R0         ;;LOAD R0 WITH TEST NUMBER
3671  010556  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3672  010562  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3673  010566  012705  064032                  MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
3674  010572  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3675  010574  000257                          CCC                     ;SCOPE SYNC
3676
3677  010576  116537  000006  063313  2$:     MOVB    6(R5),@#MBUF0+1 ;TEST THE MOVB
3678
3679  010604  020412                          CMP     R4,(R2)         ;RESULT OK?
3680  010606  001402                          BEQ     TST121          ;;BR IF YES
3681
3682  010610  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3683  010612  000767                          BR      1$              ;LOCK ON HARD ERRROR
3684                                  ;;**********************************************************************
3685                                  ;*TEST 121       BASIC 'MOVB X(R),@#A'' TEST - SRC ODD / DEST ODD
3686                                  ;;**********************************************************************
3687  010614                          TST121:
3688  010614  012700  000121                  MOV     #121,R0         ;;LOAD R0 WITH TEST NUMBER
3689                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT2
3690  010620  032737  000004  063234          BIT     #BIT2,@#BPTIOC  ;BREAKPOINT HALT SET ??
3691  010626  001401                          BEQ     .+4             ;BR IF NOT
3692  010630  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
3693  010632  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3694  010636  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3695  010642  012705  064634                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3696  010646  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
```

```
3697  010650  000257                        CCC                     ;SCOPE SYNC
3698
3699  010652  116537  000001  063313  2$:   MOVB    1(R5),@#MBUF0+1 ;TEST THE MOVB
3700
3701  010660  020412                        CMP     R4,(R2)         ;RESULT OK?
3702  010662  001402                        BEQ     TST122          ;;BR IF YES
3703
3704  010664  000000                  3$:   HALT                    ;MOVB DELIVERED WRONG RESULT
3705  010666  000767                        BR      1$              ;LOCK ON HARD ERRROR
3706
3707                                  ;;*****************************************************************
3708                                  ;*TEST 122       BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=0
3709                                  ;;*****************************************************************
3710  010670                          TST122:
3711  010670  012700  000122                MOV     #122,R0         ;;LOAD R0 WITH TEST NUMBER
3712  010674  000257                  1$:   CCC                     ;CLEAR ALL FLAGS
3713
3714  010676  001404                  2$:   BEQ     3$              ;NO BR SHOULD OCCUR-FLAG=0
3715  010700  100403                        BMI     3$              ;NO BR SHOULD OCCUR-FLAG=0
3716  010702  102402                        BVS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3717  010704  103401                        BCS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3718  010706  000402                        BR      TST123          ;;GO TO NEXT TEST
3719
3720  010710  000000                  3$:   HALT                    ;ONE OF ABOVE BR'S FAILED
3721  010712  000770                        BR      1$              ;ERROR LOOP RETURN
3722
3723                                  ;;*****************************************************************
3724                                  ;*TEST 123       BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=1
3725                                  ;;*****************************************************************
3726  010714                          TST123:
3727  010714  012700  000123                MOV     #123,R0         ;;LOAD R0 WITH TEST NUMBER
3728  010720  000277                  1$:   SCC                     ;MAKE N:C = 1111
3729
3730  010722  001402                  21$:  BEQ     22$             ;TEST THE BEQ-IT SHOULD BR
3731
3732  010724  000000                  3$:   HALT                    ;BEQ FAILED
3733  010726  000774                        BR      1$              ;ERROR LOOP RETURN
3734
3735  010730  100402                  22$:  BMI     23$             ;TEST THE BMI-IT SHOULD BR
3736
3737  010732  000000                  5$:   HALT                    ;BMI FAILED
3738  010734  000771                        BR      1$              ;ERROR LOOP RETURN
3739
3740  010736  102402                  23$:  BVS     24$             ;TEST THE BVS-IT SHOULD BR
3741
3742  010740  000000                  7$:   HALT                    ;BVS FAILED
3743  010742  000766                        BR      1$              ;ERROR LOOP RETURN
3744
3745  010744                          24$:
3746  010744  103402                        BCS     TST124          ;;TEST THE BCS-IT SHOULD BR
3747
3748  010746  000000                  9$:   HALT                    ;BCS FAILED
3749  010750  000763                        BR      1$              ;ERROR LOOP RETURN
3750
3751                                  ;;*****************************************************************
3752                                  ;*TEST 124       BASIC BVC TEST WITH V=1
```

```
                                        ;;*****************************************************************
010752                                  TST124:
010752  012700  000124                          MOV     #124,R0         ;;LOAD R0 WITH TEST NUMBER

010756  000262                          1$:     SEV                     ;MAKE V=1

010760  102001                          2$:     BVC     3$              ;TEST THE BVC-IT SHOULDN'T BR
010762  000402                                  BR      TST125          ;;GO TO NEXT TEST

010764  000000                          3$:     HALT                    ;BVC FAILED
010766  000773                                  BR      1$              ;ERROR LOOP RETURN

                                        ;;*****************************************************************
                                        ;*TEST 125      BASIC BVC TEST WITH V=0
                                        ;;*****************************************************************
010770                                  TST125:
010770  012700  000125                          MOV     #125,R0         ;;LOAD R0 WITH TEST NUMBER

010774  000242                          1$:     CLV                     ;MAKE V=0

010776                                  2$:
010776  102002                                  BVC     TST126          ;;TEST THE BVC-IT SHOULD BR

011000  000000                          3$:     HALT                    ;BVC FAILED
011002  000774                                  BR      1$              ;ERROR LOOP RETURN

                                        ;;*****************************************************************
                                        ;*TEST 126      BASIC BGE TEST WITH N,V = 00
                                        ;;*****************************************************************
011004                                  TST126:
011004  012700  000126                          MOV     #126,R0         ;;LOAD R0 WITH TEST NUMBER

011010  000257                          1$:     CCC                     ;MAKE N:C = 0000

011012                                  2$:
011012  002002                                  BGE     TST127          ;;TEST THE BGE-IT SHOULD BR

011014  000000                          3$:     HALT                    ;BGE FAILED
011016  000774                                  BR      1$              ;ERROR LOOP RETURN

                                        ;;*****************************************************************
                                        ;*TEST 127      BASIC BGE TEST WITH N,V = 01
                                        ;;*****************************************************************
011020                                  TST127:
011020  012700  000127                          MOV     #127,R0         ;;LOAD R0 WITH TEST NUMBER

011024  000257                          1$:     CCC                     ;CLEAR FLAGS
011026  000262                                  SEV                     ;MAKE N,V = 01

011030  002001                          2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
011032  000402                                  BR      TST130          ;;GO TO NEXT TEST

011034  000000                          3$:     HALT                    ;BGE FAILED
011036  000772                                  BR      1$              ;ERROR LOOP RETURN

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

```
3809                                ;*TEST 130      BASIC BGE TEST WITH N,V = 10
3810                                ;;*****************************************************************
3811  011040                        TST130:
3812  011040  012700  000130               MOV     #130,R0         ;;LOAD R0 WITH TEST NUMBER
3813
3814  011044  000257                1$:     CCC                     ;CLEAR FLAGS
3815  011046  000270                       SEN                     ;MAKE N,V = 10
3816
3817  011050  002001                2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3818  011052  000402                       BR      TST131          ;;GO TO NEXT TEST
3819
3820  011054  000000                3$:     HALT                    ;BGE FAILED
3821  011056  000772                       BR      1$              ;ERROR LOOP RETURN
3822
3823                                ;;*****************************************************************
3824                                ;*TEST 131      BASIC BGE TEST WITH N,V = 11
3825                                ;;*****************************************************************
3826  011060                        TST131:
3827  011060  012700  000131               MOV     #131,R0         ;;LOAD R0 WITH TEST NUMBER
3828
3829  011064  000257                1$:     CCC                     ;CLEAR FLAGS
3830  011066  000272                       272                     ;MAKE N,V = 11
3831
3832  011070                        2$:
3833  011070  002002                       BGE     TST132          ;;TEST THE BGE-IT SHOULD BR
3834
3835  011072  000000                3$:     HALT                    ;BGE FAILED
3836  011074  000773                       BR      1$              ;ERROR LOOP RETURN
3837
3838                                ;;*****************************************************************
3839                                ;*TEST 132      BASIC BLT TEST WITH N,V = 00
3840                                ;;*****************************************************************
3841  011076                        TST132:
3842  011076  012700  000132               MOV     #132,R0         ;;LOAD R0 WITH TEST NUMBER
3843
3844  011102  000257                1$:     CCC                     ;CLEAR FLAGS
3845
3846  011104  002401                2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3847  011106  000402                       BR      TST133          ;;GO TO NEXT TEST
3848
3849  011110  000000                3$:     HALT                    ;BLT FAILED
3850  011112  000773                       BR      1$              ;ERROR LOOP RETURN
3851
3852                                ;;*****************************************************************
3853                                ;*TEST 133      BASIC BLT TEST WITH N,V = 01
3854                                ;;*****************************************************************
3855  011114                        TST133:
3856  011114  012700  000133               MOV     #133,R0         ;;LOAD R0 WITH TEST NUMBER
3857
3858  011120  000257                1$:     CCC                     ;CLEAR FLAGS
3859  011122  000262                       SEV                     ;MAKE N,V = 01
3860
3861  011124                        2$:
3862  011124  002402                       BLT     TST134          ;;TEST THE BLT-IT SHOULD BR
3863
3864  011126  000000                3$:     HALT                    ;BLT FAILED
```

```
3865  011130  000773                  BR      1$              ;ERROR LOOP RETURN
3866
3867                                  ;;*********************************************************************
3868                                  ;*TEST 134       BASIC BLT TEST WITH N,V = 10
3869                                  ;;*********************************************************************
3870  011132                          TST134:
3871  011132  012700  000134          MOV     #134,R0         ;;LOAD R0 WITH TEST NUMBER
3872
3873  011136  000257                  1$:     CCC             ;CLEAR FLAGS
3874  011140  000270                  SEN                     ;SET N - N,V = 10
3875
3876  011142                          2$:
3877  011142  002402                  BLT     TST135          ;;TEST THE BLT-IT SHOULD BR
3878
3879  011144  000000                  3$:     HALT            ;BLT FAILED
3880  011146  000773                  BR      1$              ;ERROR LOOP RETURN
3881
3882                                  ;;*********************************************************************
3883                                  ;*TEST 135       BASIC BLT TEST WITH N,V = 11
3884                                  ;;*********************************************************************
3885  011150                          TST135:
3886  011150  012700  000135          MOV     #135,R0         ;;LOAD R0 WITH TEST NUMBER
3887
3888  011154  000257                  1$:     CCC             ;CLEAR FLAGS
3889  011156  000272                  272                     ;MAKE N,V = 11
3890
3891  011160  002401                  2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3892  011162  000402                  BR      TST136          ;;GO TO NEXT TEST
3893
3894  011164  000000                  3$:     HALT            ;BLT FAILED
3895  011166  000772                  BR      1$              ;ERROR LOOP RETURN
3896
3897                                  ;;*********************************************************************
3898                                  ;*TEST 136       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3899                                  ;;*********************************************************************
3900  011170                          TST136:
3901  011170  012700  000136          MOV     #136,R0         ;;LOAD R0 WITH TEST NUMBER
3902
3903  011174  000257                  1$:     CCC             ;CLEAR FLAGS
3904  011176  000266                  266                     ;SET Z AND V
3905
3906  011200  003001                  2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3907  011202  000402                  BR      TST137          ;;GO TO NEXT TEST
3908
3909  011204  000000                  3$:     HALT            ;BGT FAILED
3910  011206  000772                  BR      1$              ;ERROR LOOP RETURN
3911
3912                                  ;;*********************************************************************
3913                                  ;*TEST 137       BASIC BGT TEST WITH Z = 0 AND N,V = 01
3914                                  ;;*********************************************************************
3915  011210                          TST137:
3916  011210  012700  000137          MOV     #137,R0         ;;LOAD R0 WITH TEST NUMBER
3917
3918  011214  000257                  1$:     CCC             ;CLEAR FLAGS
3919  011216  000262                  SEV                     ;SET V
3920
```

```
3921  011220  003001          2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3922  011222  000402                  BR      TST140          ;;GO TO SCOPE LOOP EXIT
3923
3924  011224  000000          3$:     HALT                    ;BGT FAILED
3925  011226  000772                  BR      1$              ;ERROR LOOP RETURN
3926
3927                          ;;**********************************************************************
3928                          ;*TEST 140       BASIC BGT TEST WITH Z = 1 AND N,V = 00
3929                          ;;**********************************************************************
3930  011230                  TST140:
3931  011230  012700  000140          MOV     #140,R0         ;;LOAD R0 WITH TEST NUMBER
3932
3933  011234  000257          1$:     CCC                     ;CLEAR FLAGS
3934  011236  000264                  SEZ                     ;SET Z
3935
3936  011240  003001          2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3937  011242  000402                  BR      TST141          ;;GO TO SCOPE LOOP EXIT
3938
3939  011244  000000          3$:     HALT                    ;BGT FAILED
3940  011246  000772                  BR      1$              ;ERROR LOOP RETURN
3941
3942                          ;;**********************************************************************
3943                          ;*TEST 141       BASIC BGT TEST WITH Z = 0 AND N,V = 00
3944                          ;;**********************************************************************
3945  011250                  TST141:
3946  011250  012700  000141          MOV     #141,R0         ;;LOAD R0 WITH TEST NUMBER
3947
3948  011254  000257          1$:     CCC                     ;CLEAR FLAGS
3949
3950  011256                  2$:
3951  011256  003002                  BGT     TST142          ;;TEST THE BGT - IT SHOULD BR
3952
3953  011260  000000          3$:     HALT                    ;BGT FAILED
3954  011262  000774                  BR      1$              ;ERROR LOOP RETURN
3955
3956                          ;;**********************************************************************
3957                          ;*TEST 142       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3958                          ;;**********************************************************************
3959  011264                  TST142:
3960  011264  012700  000142          MOV     #142,R0         ;;LOAD R0 WITH TEST NUMBER
3961
3962  011270  000257          1$:     CCC                     ;CLEAR FLAGS
3963  011272  000266                  266                     ;MAKE N,V = 01 AND Z = 1
3964
3965  011274  003001          2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3966  011276  000402                  BR      TST143          ;;GO TO NEXT TEST
3967
3968  011300  000000          3$:     HALT                    ;BGT FAILED
3969  011302  000772                  BR      1$              ;ERROR LOOP RETURN
3970
3971                          ;;**********************************************************************
3972                          ;*TEST 143       BASIC BGT TEST WITH Z = 1 AND N,V = 10
3973                          ;;**********************************************************************
3974  011304                  TST143:
3975  011304  012700  000143          MOV     #143,R0         ;;LOAD R0 WITH TEST NUMBER
3976
```

```
3977  011310  000257                  1$:     CCC                     ;CLEAR FLAGS
3978  011312  000274                          274                     ;MAKE Z = 1 AND N,V = 10
3979
3980  011314  003001                  2$:     BGT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3981  011316  000402                          BR      TST144          ;;GO TO NEXT TEST
3982
3983  011320  000000                  3$:     HALT                    ;BLT FAILED
3984  011322  000772                          BR      1$              ;ERROR LOOP RETURN
3985
3986                                  ;;*****************************************************************
3987                                  ;*TEST 144       BASIC BGT TEST WITH Z = 1 AND N,V = 11
3988                                  ;;*****************************************************************
3989  011324                          TST144:
3990  011324  012700  000144                 MOV     #144,R0         ;;LOAD R0 WITH TEST NUMBER
3991
3992  011330  000257                  1$:     CCC                     ;CLEAR FLAGS
3993  011332  000276                          276                     ;MAKE Z = 1 AND N,V = 11
3994
3995  011334  003001                  2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3996  011336  000402                          BR      TST145          ;;GO TO NEXT TEST
3997
3998  011340  000000                  3$:     HALT                    ;BLT FAILED
3999  011342  000772                          BR      1$              ;ERROR LOOP RETURN
4000
4001                                  ;;*****************************************************************
4002                                  ;*TEST 145       BASIC BGT TEST WITH Z=0 AND N,V=11
4003                                  ;;*****************************************************************
4004  011344                          TST145:
4005  011344  012700  000145                 MOV     #145,R0         ;;LOAD R0 WITH TEST NUMBER
4006
4007  011350  000257                  1$:     CCC                     ;CLEAR FLAGS
4008  011352  000272                          272                     ;MAKE N:C=1010
4009
4010  011354                          2$:
4011  011354  003002                          BGT     TST146          ;;TEST THE BGT - IT SHOULD BR
4012
4013  011356  000000                  3$:     HALT                    ;BGT FAILED
4014  011360  000773                          BR      1$              ;ERROR LOOP RETURN
4015
4016                                  ;;*****************************************************************
4017                                  ;*TEST 146       BASIC BHI TEST WITH Z,C = 00
4018                                  ;;*****************************************************************
4019  011362                          TST146:
4020  011362  012700  000146                 MOV     #146,R0         ;;LOAD R0 WITH TEST NUMBER
4021
4022  011366  000257                  1$:     CCC                     ;MAKE Z,C = 00
4023
4024  011370                          2$:
4025  011370  101002                          BHI     TST147          ;;TEST THE BHI-IT SHOULD BR
4026
4027  011372  000000                  3$:     HALT                    ;BHI FAILED
4028  011374  000774                          BR      1$              ;ERROR LOOP RETURN
4029
4030                                  ;;*****************************************************************
4031                                  ;*TEST 147       BASIC BHI TEST WITH Z,C = 01
4032                                  ;;*****************************************************************
```

```
 4033  011376
 4034  011376  012700  000147          TST147:
                                               MOV     #147,R0         ;;LOAD R0 WITH TEST NUMBER
 4035
 4036  011402  000257                  1$:     CCC                     ;CLEAR FLAGS
 4037  011404  000261                          SEC                     ;MAKE Z,C = 01
 4038
 4039  011406  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4040  011410  000402                          BR      TST150          ;;GO TO NEXT TEST
 4041
 4042  011412  000000                  3$:     HALT                    ;BHI FAILED
 4043  011414  000772                          BR      1$              ;ERROR LOOP RETURN
 4044
 4045                                  ;;*****************************************************************
 4046                                  ;*TEST 150      BASIC BHI TEST WITH Z,C = 10
 4047                                  ;;*****************************************************************
 4048  011416                          TST150:
 4049  011416  012700  000150                 MOV     #150,R0         ;;LOAD R0 WITH TEST NUMBER
 4050
 4051  011422  000257                  1$:     CCC                     ;CLEAR FLAGS
 4052  011424  000264                          SEZ                     ;MAKE Z,C = 10
 4053
 4054  011426  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4055  011430  000402                          BR      TST151          ;;GO TO NEXT TEST
 4056
 4057  011432  000000                  3$:     HALT                    ;BHI FAILED
 4058  011434  000772                          BR      1$              ;ERROR LOOP RETURN
 4059
 4060                                  ;;*****************************************************************
 4061                                  ;*TEST 151      BASIC BHI TEST WITH Z,C = 11
 4062                                  ;;*****************************************************************
 4063  011436                          TST151:
 4064  011436  012700  000151                 MOV     #151,R0         ;;LOAD R0 WITH TEST NUMBER
 4065
 4066  011442  000257                  1$:     CCC                     ;CLEAR FLAGS
 4067  011444  000265                          265                     ;MAKE Z,C = 11
 4068
 4069  011446  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULDN'T BR
 4070  011450  000402                          BR      TST152          ;;GO TO NEXT TEST
 4071
 4072  011452  000000                  3$:     HALT                    ;BHI FAILED
 4073  011454  000772                          BR      1$              ;ERROR LOOP RETURN
 4074
 4075                                  ;;*****************************************************************
 4076                                  ;*TEST 152      BASIC NEG MODE 0 TEST : [DEST] GT 0
 4077                                  ;;*****************************************************************
 4078  011456                          TST152:
 4079  011456  012700  000152                 MOV     #152,R0         ;;LOAD R0 WITH TEST NUMBER
 4080  011462  012704  177776                 MOV     #-2,R4          ;RESULT S / B = 177776
 4081  011466  012703  000002          1$:    MOV     #2,R3           ;INITIAL [DEST] = 2
 4082  011472  000257                          CCC                     ;CLEAR FLAGS
 4083  011474  000266                          266                     ;MAKE N:C = 0110
 4084
 4085  011476  005403                  2$:     NEG     R3              ;TEST THE NEG
 4086
 4087  011500  100003                          BPL     3$
 4088  011502  001402                          BEQ     3$              ;DID N:C = 1001?
```

```
 4089  011504  102401                    BVS     3$
 4090  011506  103402                    BCS     4$
 4091
 4092  011510  000000              3$:   HALT                    ;NEGATE FAILED TO ALTER CODES PROPERLY
 4093  011512  000765                    BR      1$              ;ERROR LOOP RETURN
 4094
 4095  011514  020304              4$:   CMP     R3,R4           ;CORRECT RESULT?
 4096  011516  001402                    BEQ     TST153          ;;BR IF YES
 4097
 4098  011520  000000              5$:   HALT                    ;NEG DELIVERED WRONG RESULT
 4099  011522  000761                    BR      1$              ;ERROR LOOP RETURN
 4100
 4101                              ;;*****************************************************************
 4102                              ;*TEST 153       BASIC ''SUB #,@#'' TEST
 4103                              ;;*****************************************************************
 4104  011524                      TST153:
 4105  011524  012700  000153            MOV     #153,R0         ;;LOAD R0 WITH TEST NUMBER
 4106  011530  012704  000002            MOV     #2,R4           ;RESULT S / B = 2
 4107  011534  012702  063312            MOV     #MBUF0,R2       ;R2 POINTS TO DEST
 4108  011540  012712  000004      1$:   MOV     #4,(R2)         ;INITIAL [DEST] = 4
 4109  011544  000257                    CCC                     ;CLEAR FLAGS
 4110
 4111  011546  162737  000002  063312 2$:   SUB     #2,@#MBUF0      ;TEST THE SUB
 4112
 4113  011554  020412                    CMP     R4,(R2)         ;RESULT=2?
 4114  011556  001403                    BEQ     TST154          ;;BR IF YES
 4115  011560  011203                    MOV     (R2),R3         ;GET WAS DATA
 4116  011562  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4117  011564  000765                    BR      1$              ;ERROR LOOP RETURN
 4118
 4119                              ;;*****************************************************************
 4120                              ;*TEST 154       BASIC ''SUB @#,RN'' TEST
 4121                              ;;*****************************************************************
 4122  011566                      TST154:
 4123  011566  012700  000154            MOV     #154,R0         ;;LOAD R0 WITH TEST NUMBER
 4124  011572  012737  000002  063312    MOV     #2,@#MBUF0      ;SRC = 2
 4125  011600  012703  000004      1$:   MOV     #4,R3           ;INITIAL [DEST] = 4
 4126  011604  000257                    CCC                     ;SCOPE SYNC
 4127
 4128  011606  163703  063312      2$:   SUB     @#MBUF0,R3      ;TEST THE SUB
 4129
 4130  011612  020403                    CMP     R4,R3           ;RESULT=2?
 4131  011614  001402                    BEQ     TST155          ;;BR IF YES
 4132
 4133  011616  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4134  011620  000767                    BR      1$              ;ERROR LOOP RETURN
 4135
 4136                              ;;*****************************************************************
 4137                              ;*TEST 155       BASIC 'RTS RN'' TEST - <N:C> = 1111
 4138                              ;;*****************************************************************
 4139  011622                      TST155:
 4140  011622  012700  000155            MOV     #155,R0         ;;LOAD R0 WITH TEST NUMBER
 4141  011626  010605                    MOV     SP,R5           ;SAVE SP
 4142  011630  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4143  011632  012703  011652            MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
 4144  011636  012746  177777            MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 FROM STACK
```

```
4145  011642  000277                    SCC                     ;N:C = 1111
4146
4147  011644  000203          2$:       RTS     R3              ;TEST THE RTS - GO TO 4$
4148
4149  011646  000000          3$:       HALT                    ;RTS FAILED TO LOAD THE PC
4150  011650  000767                    BR      1$              ;LOCK ON ERROR
4151
4152  011652  100003          4$:       BPL     5$              ;N:C = 1111 ?
4153  011654  001002                    BNE     5$
4154  011656  102001                    BVC     5$
4155  011660  103402                    BCS     6$
4156
4157  011662  000000          5$:       HALT                    ;RTS ALTERED CODES - CLEARED ONE
4158  011664  000761                    BR      1$              ;LOCK ON ERROR
4159
4160  011666  020327  177777  6$:       CMP     R3,#-1          ;DID R3 GET LOADED FROM STACK ?
4161  011672  001402                    BEQ     8$              ;BR IF YES
4162
4163  011674  000000          7$:       HALT                    ;RTS FAILED TO LOAD REG
4164  011676  000754                    BR      1$              ;LOCK ON ERROR
4165
4166  011700  020506          8$:       CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
4167  011702  001402                    BEQ     TST156          ;;BR IF YES
4168
4169  011704  000000          9$:       HALT                    ;RTS FAILED TO POP SP
4170  011706  000750                    BR      1$              ;LOCK ON ERROR
4171
4172                        ;;**********************************************************************
4173                        ;*TEST 156      BASIC 'RTS  PC'' TEST
4174                        ;;**********************************************************************
4175  011710                TST156:
4176  011710  012700  000156           MOV     #156,R0         ;;LOAD R0 WITH TEST NUMBER
4177  011714  010605                    MOV     SP,R5           ;SAVE THE ORIGINAL SP
4178  011716  010506          1$:       MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4179  011720  012746  011734           MOV     #4$,-(SP)       ;PUSH NEW PC ON STACK
4180  011724  000257                    CCC                     ;SCOPE SYNC
4181
4182  011726  000207          2$:       RTS     PC              ;TEST THE RTS - GO TO 4$
4183
4184  011730  000000          3$:       HALT                    ;RTS FAILED TO LOAD PC
4185  011732  000771                    BR      1$              ;LOCK ON HARD ERROR
4186
4187  011734  020605          4$:       CMP     SP,R5           ;DID SP GET POPPED ?
4188  011736  001402                    BEQ     TST157          ;;BR IF YES
4189
4190  011740  000000          5$:       HALT                    ;RTS FAILED TO UPDATE SP
4191  011742  000765                    BR      1$              ;LOCK ON HARD ERROR
4192
4193                        ;;**********************************************************************
4194                        ;*TEST 157      BASIC ''JSR  PC,@#A'' TEST
4195                        ;;**********************************************************************
4196  011744                TST157:
4197  011744  012700  000157           MOV     #157,R0         ;;LOAD R0 WITH TEST NUMBER
4198                      .SBTTL USER CONTROLLED BREAKPOINT -- BIT3
4199  011750  032737  000010  063234   BIT     #BIT3,@#BPTLOC  ;BREAKPOINT HALT SET ??
4200  011756  001401                    BEQ     .+4             ;BR IF NOT
```

```
                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
                        MOV     SP,R5           ;SAVE ORIGINAL SP
                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                        CCC                     ;SCOPE SYNC

                2$:     JSR     PC,@#4$ ;TEST THE JSR - GO TO 4$

                3$:     HALT                    ;JSR FAILED TO LOAD PC
                        BR      1$              ;LOCK ON HARD ERROR

                4$:     CMP     #3$,(SP)+       ;DID JSR SAVE OLD PC ON STACK ?
                        BEQ     TST160          ;;BR IF YES

                5$:     HALT                    ;JSR FAILED TO SAVE OLD PC
                        BR      1$              ;LOCK ON HARD ERROR

                ;;*********************************************************************
                ;*TEST 160      BASIC 'RTI'' TEST - N:C=0000
                ;;*********************************************************************
012012          TST160:
012012  012700  000160          MOV     #160,R0         ;;LOAD R0 WITH TEST NUMBER
012016  010605                  MOV     SP,R5           ;SAVE THE SP
012020  010506          1$:     MOV     R5,SP           ;RESET THE SP FOR ERROR LOOP
012022  012746  000357          MOV     #357,-(SP)      ;NEW PSW = 357
012026  012746  012046          MOV     #4$,-(SP)       ;NEW PC = 4$
012032  005037  177776          CLR     @#PSW           ;MAKE [PSW] = 000
012036  000257                  CCC                     ;MAKE N:C=0000

012040  000002          2$:     RTI                     ;TEST THE RTI - GO TO 4$

012042  000000          3$:     HALT                    ;RTI FAILED TO LOAD PC
012044  000765                  BR      1$              ;LOOP ON HARD ERROR

012046  013702  177776  4$:     MOV     @#PSW,R2        ;SAVE THE [PSW] IN R2
012052  022702  000357          CMP     #357,R2         ;WAS [PSW] = 357 ?
012056  001404                  BEQ     6$              ;BR IF YES

012060  010237  177776          MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
012064  000000          5$:     HALT                    ;RTI FAILED TO LOAD PSW
012066  000754                  BR      1$              ;LOCK ON HARD ERROR

012070  020605          6$:     CMP     SP,R5           ;DID SP GET UPDATED OK ?
012072  001402                  BEQ     TST161          ;;BR IF YES

012074  000000          7$:     HALT                    ;RTI FAILED TO UPDATE THE SP
012076  000750                  BR      1$              ;LOCK ON HARD ERROR

                ;;*********************************************************************
                ;*TEST 161      BASIC 'RTI'' TEST WITH N:C=1111
                ;;*********************************************************************
012100          TST161:
012100  012700  000161          MOV     #161,R0         ;;LOAD R0 WITH TEST NUMBER
012104  010605                  MOV     SP,R5           ;SAVE THE SP IN R5
012106  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012110  005046                  CLR     -(SP)           ;NEW PSW = 000000
012112  012746  012130          MOV     #4$,-(SP)       ;NEW PC = 4$
```

```
4257  012116  012737  000357  177776          MOV     #357,@#PSW      ;MAKE OLD PSW = 357
4258  012124  000240                          NOP                     ;SCOPE SYNC
4259
4260  012126  000002                  2$:     RTI                     ;TEST THE RTI - GO TO 4$
4261
4262  012130  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4263  012134  022702  000000                  CMP     #0,R2           ;WAS [PSW]=000
4264  012140  001404                          BEQ     TST162          ;;BR IF YES
4265
4266  012142  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4267  012146  000000                  3$:     HALT                    ;RTI FAILED TO CLEAR PSW
4268  012150  000756                          BR      1$              ;LOCK ON HARD ERROR
4269
4270                                  ;;**********************************************************************
4271                                  ;*TEST 162       BASIC ''IOT'' TEST -VERIFY LOADING PSW WITH 357
4272                                  ;;**********************************************************************
4273  012152                          TST162:
4274  012152  012700  000162                  MOV     #162,R0         ;;LOAD R0 WITH TEST NUMBER
4275  012156  010605                          MOV     SP,R5           ;SAVE THE SP
4276  012160  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4277  012162  012737  012220  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
4278  012170  012737  000357  000022          MOV     #357,@#22
4279  012176  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
4280  012204  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
4281  012210  000257                          CCC                     ;SCOPE SYNC
4282
4283  012212  000004                  2$:     IOT                     ;TEST THE IOT
4284
4285  012214  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
4286  012216  000760                          BR      1$              ;LOCK ON HARD ERROR
4287
4288  012220  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4289  012224  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
4290  012230  001404                          BEQ     6$              ;BR IF YES
4291
4292  012232  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4293  012236  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
4294  012240  000747                          BR      1$              ;LOCK ON HARD ERROR
4295
4296  012242  022726  012214          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
4297  012246  001404                          BEQ     8$              ;BR IF YES
4298
4299  012250  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4300  012254  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
4301  012256  000740                          BR      1$              ;LOCK ON HARD ERROR
4302
4303  012260  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
4304  012262  001404                          BEQ     TST163          ;;BR IF YES
4305
4306  012264  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4307  012270  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
4308  012272  000732                          BR      1$              ;LOCK ON HARD ERROR
4309
4310                                  ;;**********************************************************************
4311                                  ;*TEST 163       BASIC ''IOT'' TEST - VERIFY LINKAGE TO SCOPE SERVICE
4312                                  ;;**********************************************************************
```

```
 4313  012274
 4314  012274  012700  000163          TST163:
 4315  012300  010605                          MOV     #163,R0         ;;LOAD R0 WITH TEST NUMBER
 4316  012302  010506                          MOV     SP,R5           ;SAVE SP
 4317  012304  005037  063244          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4318  012310  012737  061612  000020          CLR     @#SCOFLG        ;TRAP SERVICE WILL COM ''SCOFLG''
 4319  012316  005037  000022                  MOV     #SCOPEA,@#20    ;SET UP IOT VECTOR
 4320  012322  000257                          CLR     @#22
 4321                                          CCC                     ;SCOPE SYNC
 4322  012324  000004                  2$:     SCOPE                   ;TEST THE IOT
 4323
 4324  012326  005137  063244                  COM     @#SCOFLG        ;SCOFLG SHOULD BECOME 000000
 4325  012332  001402                          BEQ     4$              ;BR IF IT DID
 4326
 4327  012334  000000                  3$:     HALT                    ;IOT FAILED TO LINK TO SCOPE SERVICE
 4328  012336  000761                          BR      1$              ;LOCK ON HARD ERROR
 4329
 4330  012340  010506                  4$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
 4331                                          ;;*****************************************************************
 4332                                          ;*TEST 164       BASIC ''IOT'' TEST -VERIFY LOADING PSW WITH 357
 4333                                          ;;*****************************************************************
 4334  012342                          TST164:
 4335  012342  012700  000164                  MOV     #164,R0         ;;LOAD R0 WITH TEST NUMBER
 4336  012346  010605                          MOV     SP,R5           ;SAVE THE SP
 4337  012350  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4338  012352  012737  012410  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
 4339  012360  012737  000357  000022          MOV     #357,@#22
 4340  012366  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
 4341  012374  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
 4342  012400  000257                          CCC                     ;SCOPE SYNC
 4343
 4344  012402  000004                  2$:     IOT                     ;TEST THE IOT
 4345
 4346  012404  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
 4347  012406  000760                          BR      1$              ;LOCK ON HARD ERROR
 4348
 4349  012410  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
 4350  012414  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
 4351  012420  001404                          BEQ     6$              ;BR IF YES
 4352
 4353  012422  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4354  012426  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
 4355  012430  000747                          BR      1$              ;LOCK ON HARD ERROR
 4356
 4357  012432  022726  012404          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
 4358  012436  001404                          BEQ     8$              ;BR IF YES
 4359
 4360  012440  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4361  012444  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
 4362  012446  000740                          BR      1$              ;LOCK ON HARD ERROR
 4363
 4364  012450  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
 4365  012452  001404                          BEQ     TST165          ;;BR IF YES
 4366
 4367  012454  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4368  012460  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
```

```
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 165       BASIC IOT TEST - VERIFY LOADING PSW WITH 000
                                ;;**********************************************************************
                                TST165:
                                        MOV     #165,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     SP,R5           ;SAVE THE SP
                                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                                        MOV     #4$,@#20        ;SET UP IOT VECTOR
                                        CLR     @#22            ;
                                        MOV     #340,@#PSW      ;MAKE [PSW] = 340
                                        SCC                     ;MAKE N:C=1111

                                2$:     IOT                     ;TEST THE IOT

                                4$:     MOV     @#PSW,R2        ;GET THE [PSW]
                                        BEQ     6$              ;BR IF [PSW] = 000

                                        MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
                                3$:     HALT                    ;IOT FAILED TO CLEAR THE PSW
                                        BR      1$              ;LOCK ON HARD ERROR

                                6$:     MOV     R5,SP           ;RESET THE SP BEFORE CONTINUING

                                ;;**********************************************************************
                                ;*TEST 166       BASIC ''TRAP'' TEST - LINKAGE TO PRINT ROUTINE
                                ;;**********************************************************************
                                TST166:
                                        MOV     #166,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     SP,R5           ;SAVE THE SP
                                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                                        CLR     @#PRIFLG        ;INITIALIZE TEST FLAG
                                        CLR     @#36            ;SET UP THE ''TRAP'' VECTOR
                                        MOV     #PRINA,@#34
                                        CCC                     ;SCOPE SYNC

                                2$:     TYPE                    ;TEST THE TRAP

                                        MOV     #$TRAP,@#34     ;SETUP TRAP VECTOR
                                        MOV     #340,@#36
                                        COM     @#PRIFLG        ;SHOULD MAKE [PRIFLG] = 000000
                                        BEQ     TST167          ;;BR IF IT DID

                                3$:     HALT                    ;TRAP FAILED TO LINK TO PRINT SERV.
                                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**********************************************************************
                                ;*TEST 167       BASIC ''EMT'' TEST - LINKAGE TO ERROR SERVICE
                                ;;**********************************************************************
                                TST167:
                                        MOV     #167,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     SP,R5           ;SAVE THE SP
                                1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                                        MOV     #ERRA,@#30      ;SET UP THE EMT VECTOR
                                        CLR     @#32
```

```
                                CLR     @#ERRFLG        ;EMT SERVICE WILL COM [ERRFLG]
                                CCC                     ;SCOPE SYNC

                        2$:     ERROR                   ;TEST THE EMT

                                COM     @#ERRFLG        ;DID EMT SERV. COM ERRFLG?
                                BEQ     TST170          ;;BR IF YES

                        3$:     HALT                    ;EMT DID NOT LINK PROPERLY
                                BR      1$              ;LOCK ON HARD ERROR
                        ;;**********************************************************************
                        ;*TEST 170      BASIC TEST OF RSVD INSTR. TRAP LINKAGE
                        ;;**********************************************************************
                        TST170:
                                MOV     #170,R0         ;;LOAD R0 WITH TEST NUMBER
                                MOV     SP,R5           ;SAVE THE SP
                                MOV     #RSVTST,@#10    ;SET UP RSVD INSTR. TRAP VECTOR
                                MOV     #340,@#12
                        1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                                CLR     @#RSVFLG        ;INITIALIZE TEST FLAG THAT WILL GET
                                                        ;COMPLEMENTED BY TRAP SERVICE
                                CCC                     ;SCOPE SYNC

                        2$:     000007                  ;FORCE RSVD INSTR. TRAP

                                COM     @#RSVFLG                ;TEST FLAG SHOULD GO TO 000000
                                BEQ     4$              ;BR IF TRAP SPRUNG

                        3$:     HALT                    ;RSVD INSTR. TRAP FAILED
                                BR      1$              ;LOCK ON HARD ERROR

                        4$:     MOV     #RSERR,@#10     ;SET UP RSVD INSTR TRAP VECTOR TO POINT
                                MOV     #340,@#12       ;TO ERROR SERVICE ROUTINE

                        ;;**********************************************************************
                        ;*TEST 171      BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
                        ;;**********************************************************************
                        TST171:
                                MOV     #171,R0         ;;LOAD R0 WITH TEST NUMBER
                                MOV     SP,R5           ;SAVE THE SP
                                MOV     #BETST,@#4      ;SET UP THE BUS ERROR VECTOR
                                MOV     #340,@#6
                        1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
                                CLR     @#BERFLG        ;INITIALIZE TEST FLAG THAT WILL GET
                                                        ;COMPLEMENTED BY TRAP SERVICE
                                CCC                     ;SCOPE SYNC

                        2$:     TST     @#177700        ;FORCE BUS TIMEOUT USING R0 ADDR.

                                COM     @#BERFLG        ;TEST FLAG SHOULD GO TO 000000
                                BEQ     TST172          ;;BR IF TRAP SPRUNG

                        3$:     HALT                    ;BUS ERROR FAILED TO SPRING TRAP
                                BR      1$              ;LOCK ON HARD ERROR

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

```
                                ;*TEST 172      BASIC TEST FOR ACCESSING DL11 REGISTERS
                                ;;*****************************************************************
013016                          TST172:
013016  012700  000172                  MOV     #172,R0         ;;LOAD R0 WITH TEST NUMBER
013022  005067  050264                  CLR     MBUF0           ;INIT STALL COUNTER
013026  005367  050260          11$:    DEC     MBUF0           ;COUNT THE TIMER
013032  001375                          BNE     11$             ;BR IF NO TIMEOUT
013034  012737  013074  000004          MOV     #3$,@#4 ;SET UP BUS TIMEOUT VECTOR
013042  012737  000340  000006          MOV     #340,@#6
013050  010605                          MOV     SP,R5           ;SAVE TH SP
013052  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
013054  012702  177560                  MOV     #RCSR,R2        ;[R2] = STARTING DL11 ADDR.
013060  000257                          CCC                     ;SCOPE SYNC

013062  005722                  2$:     TST     (R2)+           ;REFERENCE DL11 - RCSR
013064  005722                          TST     (R2)+           ;REFERENCE DL11 - RDBR
013066  005722                          TST     (R2)+           ;REFERENCE DL11 - XCSR
013070  005712                          TST     (R2)            ;REFERENCE DL11 - XDBR

013072  000403                          BR      4$              ;GO TO NEXT TEST

013074  005742                  3$:     TST     -(R2)           ;BAD ADDRESS IN R2
013076  000000                          HALT                    ;ONE OF DL11 ADDR'S CAUSED TIME OUT
013100  000764                          BR      1$              ;LOCK ON HARD ERROR

013102  012737  061220  000004  4$:     MOV     #BERR,@#4       ;SET UP BUS ERROR VECTOR TO POINT
013110  012737  000340  000006          MOV     #340,@#6        ;TO ERROR SERVICE ROUTINE
                                ;;*****************************************************************
                                ;*TEST 173      BASIC TEST OF DL11 - XCSR - READY(1)
                                ;;*****************************************************************
013116                          TST173:
013116  012700  000173                  MOV     #173,R0         ;;LOAD R0 WITH TEST NUMBER
013122  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
013126  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
013132  005012                  1$:     CLR     (R2)            ;CLEAR [DEST]
013134  005001                          CLR     R1              ;SET UP TIMEOUT COUNTER
013136  000257                          CCC                     ;SCOPE SYNC

013140  020412                  2$:     CMP     R4,(R2)         ;TEST READY BIT - IT SHOULD BE SET

013142  001405                          BEQ     TST174          ;;BR IF IT WAS
013144  005301                          DEC     R1              ;TICK-TOCK GOES THE TIMER
013146  001374                          BNE     2$              ;BR IF NOT A TIMEOUT

013150  011203                          MOV     (R2),R3         ;GET THE WAS DATA
013152  000000                  3$:     HALT                    ;READY BIT IN XCSR FAILED ON A (0)
013154  000766                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 174      BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
                                ;;*****************************************************************
013156                          TST174:
013156  012700  000174                  MOV     #174,R0         ;;LOAD R0 WITH TEST NUMBER
013162  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
013166  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
013172  005012                  1$:     CLR     (R2)            ;CLEAR MAINT. BIT
```

```
4537  013174  000257                    CCC                     ;SCOPE SYNC
4538
4539  013176  020412            2$:     CMP     R4,(R2)         ;TEST MAINT(0)
4540
4541  013200  001403                    BEQ     TST175          ;;BR IF MAINT BIT CLEAR
4542
4543  013202  011203                    MOV     (R2),R3         ;GET THE WAS DATA
4544  013204  000000            3$:     HALT                    ;CAN'T CLEAR MAINT BIT
4545  013206  000771                    BR      1$              ;LOCK ON HARD ERROR
4546
4547                            ;;*****************************************************************
4548                            ;*TEST 175       BASIC TEST OF DL11 XCSR - MAINT BIT = 1
4549                            ;;*****************************************************************
4550  013210                    TST175:
4551  013210  012700  000175            MOV     #175,R0         ;;LOAD R0 WITH TEST NUMBER
4552  013214  012702  177564            MOV     #XCSR,R2        ;DEST ADDR = XCSR
4553  013220  012704  000204            MOV     #204,R4         ;RESULT S / B = 204
4554  013224  012712  000004    1$:     MOV     #4,(R2)         ;SET THE MAINT. BIT
4555  013230  000257                    CCC                     ;SCOPE SYNC
4556
4557  013232  020412            2$:     CMP     R4,(R2)         ;TEST MAINT.(1)
4558
4559  013234  001403                    BEQ     TST176          ;;BR IF IT WAS
4560
4561  013236  011203                    MOV     (R2),R3         ;GET THE WAS DATA
4562  013240  000000            3$:     HALT                    ;CAN'T SET MAINT BIT IN XCSR
4563  013242  000770                    BR      1$              ;LOCK O HARD ERROR
4564
4565                            ;;*****************************************************************
4566                            ;*TEST 176       BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
4567                            ;THIS ROUTINE USES THE MAINTENANCE MODE FEATURE OF THE DL11 TO
4568                            ;TURN AROUND A STRING OF 8 CHARACTERS TO THE DL11. THIS STRING CONSISTS
4569                            ;OF ALTERNATING NULL / DELETE CHARS WHICH ARE NON PRINTING. THE 8 CHARS
4570                            ;ARE OUTPUT THEN READ BACK INTO A CORE BUFFER AND THEN THE INPUT AND
4571                            ;OUTPUT CORE BUFFERS ARE CHECKED FOR EQUIVALENCE. IF AN ERROR IS DET-
4572                            ;ECTED DURING THE COMPARISON THE ROUTINE HALTS WITH THE WAS AND S / B
4573                            ;DATA IN R3 AND R4 RESPECTFULLY. A TIMER IS EMPLOYED TO PREVENT THE
4574                            ;TEST FROM HANGING IF RECEIVER DONE DOES NOT RESPOND.
4575                            ;;*****************************************************************
4576  013244                    TST176:
4577  013244  012700  000176            MOV     #176,R0         ;;LOAD R0 WITH TEST NUMBER
4578  013250  012702  177560    6$:     MOV     #RCSR,R2        ;R2 POINTS TO DL11 - START ADDR
4579  013254  105762  000002            TSTB    2(R2)           ;REFERENCE DL11 INPUT DATA BUFFER TWICE
4580  013260  105762  000002            TSTB    2(R2)           ;TO FLUSH RCVR 'DONE'' BIT
4581  013264  012703  063266            MOV     #IBUF,R3        ;R3 POINTS TO CORE INPUT BUFFER
4582  013270  012704  063256            MOV     #OBUF,R4        ;R4 POINTS TO CORE OUTPUT BUFFER
4583  013274  012705  000010            MOV     #10,R5          ;R5 WILL COUNT 8 CHARS OUTPUT
4584  013300  012762  000004  000004    MOV     #4,4(R2)        ;TURN ON MAINT MODE
4585
4586  013306  005001            1$:     CLR     R1              ;R1 USED AS TIMEOUT COUNTER
4587  013310  112462  000006            MOVB    (R4)+,6(R2)     ;LOAD OUTPUT BUFFER IN DL11
4588  013314  105712            2$:     TSTB    (R2)            ;RECEIVER DONE SET ?
4589  013316  100404                    BMI     3$              ;BR IF YES
4590  013320  005301                    DEC     R1              ;COUNT THE TIMER
4591  013322  001374                    BNE     2$              ;BR IF NO TIMEOUT
4592
```

```
013324  000000                          HALT                    ;DL11 FAILED TO RESPOND IN TIME
013326  000750                          BR      6$              ;LOCK ON HARD ERROR

013330  116223  000002          3$:     MOVB    2(R2),(R3)+     ;READ THE DL11 INPUT BUFFER INTO CORE
013334  005305                          DEC     R5              ;COUNT ONE CHAR
013336  001363                          BNE     1$              ;BR IF NOT DONE 8 CHARS

013340  005062  000004                  CLR     4(R2)           ;TURN OFF MAINT. MODE
013344  012705  000010                  MOV     #10,R5          ;RESET CHAR COUNTER
013350  012703  063266                  MOV     #IBUF,R3        ;RESET INBUF POINTER
013354  012704  063256                  MOV     #OBUF,R4        ;RESET OUTBUF POINTER

013360  122324                  4$:     CMPB    (R3)+,(R4)+     ;INPUT = OUTPUT ??
013362  001003                          BNE     5$              ;BR IF NOT
013364  005305                          DEC     R5              ;COUNT ONE CHECKED
013366  001374                          BNE     4$              ;BR UNTIL 8 DONE
013370  000410                          BR      CITST           ;GO TO NEXT TEST

013372  114303                  5$:     MOVB    -(R3),R3        ;WAS DATA IN R3 [BITS 7:0]
013374  114404                          MOVB    -(R4),R4        ;S / B DATA IN R4 [BITS 7:0]
013376  042703  177400                  BIC     #177400,R3      ;STRIP OFF BITS <15:08>
013402  042704  177400                  BIC     #177400,R4      ;  "          "
013406  000000                          HALT                    ;RECEIVED DATA NOT EQUAL TO OUTPUT DATA
013410  000717                          BR      6$              ;LOCK ON HARD ERROR
```

```
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                ;////////////COMPREHENSIVE INSTRUCTION TESTS/////////////////////
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

013412  012737  061260  000020  CITST:  MOV     #$SCOPE,@#20    ;SET UP IOT VECTOR
013420  005037  000022                  CLR     @#22
013424  012737  061620  000030          MOV     #$ERROR,@#30    ;SET UP EMT VECTOR
013432  012737  000340  000032          MOV     #340,@#32
013440  012737  063166  000034          MOV     #$TRAP,@#34     ;SET UP TRAP VECTOR
013446  012737  000340  000036          MOV     #340,@#36
013454  012737  060664  000024          MOV     #$PWRDN,@#24    ;SET UP POWER FAIL VECTOR
013462  012737  000340  000026          MOV     #340,@#26
013470  105737  001141                  TSTB    @#$ENVM         ;DO NOT SIZE BIT SET?
013474  100003                          BPL     3$              ;BR IF NOT - USE HARDWARE SWITCH REG
013476  012737  001142  001040          MOV     #$SWREG,@#SWR   ;USE APT SWITCH REG.
013504  032777  010000  165326  3$:     BIT     #SW12,@SWR      ;INHIBIT PRINTING INTRO. I.D. MESSAGE?
013512  001007                          BNE     1$              ;BR IF YES
013514  005737  063254                  TST     @#ONCE          ;FIRST TIME INTO "CIT" TESTS ?
013520  001004                          BNE     1$              ;BR IF NOT - PRINT ID ONLY ONCE
013522  005137  063254                  COM     @#ONCE          ;SET FLAG TO INHIBIT PRINTING AGAIN
013526  104401                          TYPE                    ;IDENTIFY THIS PROGRAM
013530  065141                          IDENT1                  ;ADDR OF THE ID MESSAGE
013532  005037  177776          1$:     CLR     @#PSW           ;SET CPU PRIORITY TO LEVEL 000
013536  012737  003316  001006          MOV     #TST0,@#$LPADR  ;INITIALIZE SCOPE LOOP RETURN
013544  012737  000040  001110          MOV     #40,@#$TIMES    ;ITERATE ON BIT SECTION 32 TIMES
013552  010037  001124                  MOV     R0,@#$TESTN     ;PREVENT MISSED TEST ERROR ON
                                                                ;FIRST SCOPE CALL

                                ;;*****************************************************************
                                ;*TEST 177       BCC TEST WITH C=1
                                ;;*****************************************************************
013556                          TST177:
013556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013560  012700  000177                  MOV     #177,R0         ;;LOAD R0 WITH TEST NUMBER
013564  013701  013572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013570  000261                          SEC                     ;MAKE C=1

013572  103001          2$:     BCC     3$              ;TEST THE BCC, IT SHOULDN'T BR
013574  000401                  BR      TST200          ;;GO TO SCOPE EXIT

013576  104005          3$:     ERROR   5               ;BCC FAILED

                                ;;*****************************************************************
                                ;*TEST 200       BCC TEST WITH C=0
                                ;;*****************************************************************
013600                          TST200:
013600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013602  012700  000200                  MOV     #200,R0         ;;LOAD R0 WITH TEST NUMBER
013606  013701  013614                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013612  000241                          CLC                     ;MAKE C=0

013614                          2$:
013614  103001                  BCC     TST201          ;;TEST THE BCC-IT SHOULD BR

013616  104005          3$:     ERROR   5               ;BCC FAILED
```

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

```
4729  013746  013703  177776          MOV     @#PSW,R3        ;GET FLAGS
4730  013752  005703                  TST     R3              ;N:C = 0000
4731  013754  001401                  BEQ     TST204          ;;BR IF YES
4732
4733  013756  104001          3$:     ERROR   1               ;NO BRANCH MICROROUTINE-ALTERED CODES
4734
4735                          ;;*****************************************************************
4736                          ;*TEST 204      VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
4737                          ;;*****************************************************************
4738  013760                  TST204:
4739  013760  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4740  013762  012700  000204          MOV     #204,R0         ;;LOAD R0 WITH TEST NUMBER
4741  013766  013701  014002          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4742  013772  005004                  CLR     R4              ;PSW S/B = 0
4743  013774  012702  177776          MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL
4744
4745  014000  000257                  CCC                     ;MAKE N:C = 0000
4746
4747  014002  000401          2$:     BR      4$              ;TEST THE BR
4748
4749  014004  104005          3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK
4750
4751  014006  013703  177776  4$:     MOV     @#PSW,R3        ;GET FLAGS
4752  014012  005703                  TST     R3              ;N:C = 0000
4753  014014  001401                  BEQ     TST205          ;;BR IF YES
4754
4755  014016  104001          5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES.
4756
4757                          ;;*****************************************************************
4758                          ;*TEST 205      BLE TEST WITH Z = 0, AND N,V = 00
4759                          ;;*****************************************************************
4760  014020                  TST205:
4761  014020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4762  014022  012700  000205          MOV     #205,R0         ;;LOAD R0 WITH TEST NUMBER
4763  014026  013701  014034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4764  014032  000257                  CCC                     ;CLEAR FLAGS
4765
4766  014034  003401          2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
4767  014036  000401                  BR      TST206          ;;GO TO SCOPE EXIT
4768
4769  014040  104005          3$:     ERROR   5               ;BLE FAILED
4770
4771                          ;;*****************************************************************
4772                          ;*TEST 206      BLE TEST WITH Z = 1 AND N,V = 00
4773                          ;;*****************************************************************
4774  014042                  TST206:
4775  014042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4776  014044  012700  000206          MOV     #206,R0         ;;LOAD R0 WITH TEST NUMBER
4777  014050  013701  014060          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4778  014054  000257                  CCC                     ;CLEAR FLAGS
4779  014056  000264                  SEZ                     ;SET Z = 1
4780
4781  014060          2$:
4782  014060  003401                  BLE     TST207          ;;TEST THE BLE-IT SHOULD BR
4783
4784  014062  104005          3$:     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
```

```
4841  014172  000401                    BR      TST213          ;;GO TO SCOPE EXIT
4842
4843  014174  104005            3$:     ERROR   5               ;BLOS FAILED
4844
4845                            ;;*****************************************************************
4846                            ;*TEST 213       BLOS TEST WITH Z,C = 01
4847                            ;;*****************************************************************
4848  014176                    TST213:
4849  014176  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4850  014200  012700  000213            MOV     #213,R0         ;;LOAD R0 WITH TEST NUMBER
4851  014204  013701  014214            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4852  014210  000257                    CCC                     ;CLEAR FLAGS
4853  014212  000261                    SEC                     ;MAKE Z,C = 01
4854
4855  014214                    2$:
4856  014214  101401                    BLOS    TST214          ;;TEST THE BLOS-IT SHOULD BR
4857
4858  014216  104005            3$:     ERROR   5               ;BLOS FAILED
4859
4860                            ;;*****************************************************************
4861                            ;*TEST 214       BLOS TEST WITH Z,C = 10
4862                            ;;*****************************************************************
4863  014220                    TST214:
4864  014220  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4865  014222  012700  000214            MOV     #214,R0         ;;LOAD R0 WITH TEST NUMBER
4866  014226  013701  014236            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4867  014232  000257                    CCC                     ;CLEAR FLAGS
4868  014234  000264                    SEZ                     ;MAKE Z,C = 10
4869
4870  014236                    2$:
4871  014236  101401                    BLOS    TST215          ;;TEST THE BLOS-IT SHOULD BR
4872
4873  014240  104005            3$:     ERROR   5               ;BLOS FAILED
4874
4875                            ;;*****************************************************************
4876                            ;*TEST 215       BLOS TEST WITH Z,C = 11
4877                            ;;*****************************************************************
4878  014242                    TST215:
4879  014242  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4880  014244  012700  000215            MOV     #215,R0         ;;LOAD R0 WITH TEST NUMBER
4881  014250  013701  014260            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4882  014254  000257                    CCC                     ;CLEAR FLAGS
4883  014256  000265                    265                     ;MAKE Z,C = 11
4884
4885  014260                    2$:
4886  014260  101401                    BLOS    TST216          ;;TEST THE BLOS-IT SHOULD BR
4887
4888  014262  104005            3$:     ERROR   5               ;BLOS FAILED
4889
4890                            ;;*****************************************************************
4891                            ;*TEST 216       SXT MODE 0 TEST WITH N = 0 AND C = 1
4892                            ;;*****************************************************************
4893  014264                    TST216:
4894  014264  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4895  014266  012700  000216            MOV     #216,R0         ;;LOAD R0 WITH TEST NUMBER
4896  014272  013701  014310            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 4897  014276  005004                    CLR     R4              ;RESULT S / B = 0
 4898  014300  012703  177777            MOV     #-1,R3          ;INITAL DEST. OP = 177777
 4899  014304  000257                    CCC                     ;CLEAR CODES
 4900  014306  000263                    263                     ;N:C = 0011
 4901
 4902  014310  006703                2$: SXT     R3              ;TEST THE SXT
 4903
 4904  014312  100403                    BMI     3$
 4905  014314  001002                    BNE     3$              ;DID SXT MAKE N:C = 0101?
 4906  014316  102401                    BVS     3$
 4907  014320  103401                    BCS     4$
 4908
 4909  014322  104002                3$: ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
 4910
 4911  014324  005703                4$: TST     R3              ;DID RESULT = 0?
 4912  014326  001401                    BEQ     TST217          ;;BR IF IT DID
 4913
 4914  014330  104002                5$: ERROR   2               ;SXT DELIVERED WRONG RESULT TO R3
 4915
 4916                                    ;;*************************************************************
 4917                                    ;*TEST 217      SXT MODE 0 TEST WITH N = 0 AND C = 0
 4918                                    ;;*************************************************************
 4919  014332                        TST217:
 4920  014332  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4921  014334  012700  000217            MOV     #217,R0         ;;LOAD R0 WITH TEST NUMBER
 4922  014340  013701  014366            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4923                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT4
 4924  014344  032737  000020  063234    BIT     #BIT4,@#BPTLOC  ;BREAKPOINT HALT SET ??
 4925  014352  001401                    BEQ     .+4             ;BR IF NOT
 4926  014354  000000                    HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 4927
 4928  014356  005004                    CLR     R4              ;RESULT S / B = 0
 4929  014360  012703  177777            MOV     #-1,R3          ;INITIAL DEST OP = 177777
 4930  014364  000257                    CCC                     ;CLEAR N:C
 4931
 4932  014366  006703                2$: SXT     R3              ;TEST THE SXT
 4933  014370  103001                    BCC     TST220          ;;BR IF "C" STILL CLEAR
 4934
 4935  014372  104002                3$: ERROR   2               ;SXT AFFECTED "C" BIT
 4936
 4937                                    ;;*************************************************************
 4938                                    ;*TEST 220      SXT MODE 0 TEST WITH N = 1 AND C = 1
 4939                                    ;;*************************************************************
 4940  014374                        TST220:
 4941  014374  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4942  014376  012700  000220            MOV     #220,R0         ;;LOAD R0 WITH TEST NUMBER
 4943  014402  013701  014416            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4944  014406  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
 4945  014412  005003                    CLR     R3              ;INITIAL DEST OP = 0
 4946  014414  000277                    SCC                     ;MAKE N:C = 1111
 4947
 4948  014416  006703                2$: SXT     R3              ;TEST THE SXT
 4949
 4950  014420  100003                    BPL     3$
 4951  014422  001402                    BEQ     3$              ;N:C = 1001?
 4952  014424  102401                    BVS     3$
```

```
4953  014426  103401                  BCS     4$
4954
4955  014430  104002          3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4956
4957  014432  010305          4$:     MOV     R3,R5           ;GET RESULT
4958  014434  005105                  COM     R5              ;COMPLEMENT IT-SHOULD GO TO 0
4959  014436  001401                  BEQ     TST221          ;;BR IF RESULT OF SXT = 1
4960
4961  014440  104002          5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT.
4962
4963                          ;;*******************************************************************
4964                          ;*TEST 221       SXT MODE 0 TEST WITH N = 1 AND C = 0
4965                          ;;*******************************************************************
4966  014442                  TST221:
4967  014442  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4968  014444  012700  000221          MOV     #221,R0         ;;LOAD R0 WITH TEST NUMBER
4969  014450  013701  014466          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4970  014454  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
4971  014460  005003                  CLR     R3              ;INITIAL DEST OP = 0
4972  014462  000257                  CCC                     ;CLEAR FLAGS
4973  014464  000276                  276                     ;MAKE N:C = 1110
4974
4975  014466  006703          2$:     SXT     R3              ;TEST THE SXT
4976  014470  103001                  BCC     TST222          ;;BR IF "C" UNAFFECTED
4977
4978  014472  104002          3$:     ERROR   2               ;SXT SET "C" BIT
4979
4980                          ;;*******************************************************************
4981                          ;*TEST 222       SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
4982                          ;;*******************************************************************
4983  014474                  TST222:
4984  014474  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4985  014476  012700  000222          MOV     #222,R0         ;;LOAD R0 WITH TEST NUMBER
4986  014502  013701  014524          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4987  014506  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4988  014512  005004                  CLR     R4              ;RESULT S / B = 0
4989  014514  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4990  014520  000257                  CCC                     ;CLEAR CODES
4991  014522  000263                  263                     ;MAKE N:C = 0011
4992
4993  014524  006712          2$:     SXT     (R2)            ;TEST THE SXT - DM1
4994
4995  014526  100403                  BMI     3$
4996  014530  001002                  BNE     3$              ;N:C = 0101
4997  014532  102401                  BVS     3$
4998  014534  103401                  BCS     4$
4999
5000  014536  104001          3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5001
5002  014540  005712          4$:     TST     (R2)            ;DID RESULT = 0?
5003  014542  001401                  BEQ     11$             ;BR IF YES
5004
5005  014544  104001          5$:     ERROR   1               ;SXT SHOULD HAVE ZEROED [DEST]
5006
5007  014546  012702  063312  11$:    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5008  014552  013701  014566          MOV     @#12$,R1        ;LOAD R1 WITH TEST INSTR WORD
```

```
5009  014556  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5010  014562  000257                  CCC                     ;CLEAR CODES
5011  014564  000263                  263                     ;MAKE N:C = 0011
5012
5013  014566  006722          12$:    SXT     (R2)+           ;TEST SXT - DM2
5014
5015  014570  100403                  BMI     7$              ;N:C = 0101 ?
5016  014572  001002                  BNE     7$
5017  014574  102401                  BVS     7$
5018  014576  103401                  BCS     6$
5019
5020  014600  104001          7$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5021
5022  014602  005737  063312  6$:     TST     @#MBUF0         ;DID RESULT GET ZEROED ?
5023  014606  001401                  BEQ     8$              ;BR IF YES
5024
5025  014610  104001          9$:     ERROR   1               ;SXT FAILED TO ZERO [DEST]
5026
5027  014612  020227  063314  8$:     CMP     R2,#MBUF0+2     ;WAS IT REALLY MODE 2 ?
5028  014616  001401                  BEQ     TST223          ;;BR IF YES
5029
5030  014620  104001                  ERROR   1               ;SXT FAILED TO AUTO INCREMENT
5031
5032                          ;;*****************************************************************
5033                          ;*TEST 223       SXT MODE 1 TEST WITH N = 0 AND C = 0
5034                          ;;*****************************************************************
5035  014622                  TST223:
5036  014622  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5037  014624  012700  000223          MOV     #223,R0         ;;LOAD R0 WITH TEST NUMBER
5038  014630  013701  014650          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5039  014634  005004                  CLR     R4              ;RESULT S / B = 0
5040  014636  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5041  014642  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5042  014646  000257                  CCC                     ;CLEAR "C" BIT
5043
5044  014650  006712          2$:     SXT     (R2)            ;TEST THE SXT
5045  014652  103001                  BCC     TST224          ;;BR IF "C" UNDISTURBED
5046
5047  014654  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
5048
5049                          ;;*****************************************************************
5050                          ;*TEST 224       SXT MODE 1 TEST WITH N = 1 AND C = 1
5051                          ;;*****************************************************************
5052  014656                  TST224:
5053  014656  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5054  014660  012700  000224          MOV     #224,R0         ;;LOAD R0 WITH TEST NUMBER
5055  014664  013701  014704          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5056  014670  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5057  014674  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5058  014700  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5059  014702  000277                  SCC                     ;MAKE N:C = 1111
5060
5061  014704  006712          2$:     SXT     (R2)            ;TEST THE SXT
5062
5063  014706  100003                  BPL     3$
5064  014710  001402                  BEQ     3$              ;N:C = 1001?
```

```
5065  014712  102401                  BVS     3$
5066  014714  103401                  BCS     4$
5067
5068  014716  104001          3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5069
5070  014720  021204          4$:     CMP     (R2),R4         ;RESULT = 177777?
5071  014722  001401                  BEQ     TST225          ;;BR IF YES
5072
5073  014724  104001          5$:     ERROR   1               ;SXT DELIVERED WRONG RESULT
5074
5075                          ;;*******************************************************************
5076                          ;*TEST 225       SXT MODE 1 TEST WITH N = 1 AND C = 0
5077                          ;;*******************************************************************
5078  014726                  TST225:
5079  014726  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5080  014730  012700  000225          MOV     #225,R0         ;;LOAD R0 WITH TEST NUMBER
5081  014734  013701  014756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5082  014740  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5083  014744  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5084  014750  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5085  014752  000257                  CCC                     ;CLEAR FLAGS
5086  014754  000276                  276                     ;MAKE N:C = 1110
5087
5088  014756  006712          2$:     SXT     (R2)            ;TEST THE SXT
5089  014760  103001                  BCC     TST226          ;;BR IF ''C'' UNAFFECTED
5090
5091  014762  104001          3$:     ERROR   1               ;SXT SET THE ''C'' BIT
5092
5093                          ;;*******************************************************************
5094                          ;*TEST 226       SWAB MODE 0 TEST WITH POS. RESULT
5095                          ;;*******************************************************************
5096  014764                  TST226:
5097  014764  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5098  014766  012700  000226          MOV     #226,R0         ;;LOAD R0 WITH TEST NUMBER
5099  014772  013701  015012          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5100  014776  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
5101  015002  012703  000377          MOV     #377,R3         ;INITIAL DEST OP = 377
5102  015006  000257                  CCC                     ;CLEAR FLAGS
5103  015010  000273                  273                     ;MAKE N:C = 1011
5104
5105  015012  000303          2$:     SWAB    R3              ;TEST THE SWAB
5106
5107  015014  100403                  BMI     3$
5108  015016  001002                  BNE     3$              ;N:C = 0100
5109  015020  102401                  BVS     3$
5110  015022  103001                  BCC     4$
5111
5112  015024  104002          3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
5113
5114  015026  020403          4$:     CMP     R4,R3           ;CORRECT RESULT?
5115  015030  001401                  BEQ     TST227          ;;BR IF YES
5116
5117  015032  104002          5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
5118
5119                          ;;*******************************************************************
5120                          ;*TEST 227       SWAB MODE 0 TEST WITH NEG. RESULT
```

```
5121
5122  015034
5123  015034  000004
5124  015036  012700  000227          ;;*****************************************************************
5125  015042  013701  015062          TST227:
5126  015046  012704  000377                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5127  015052  012703  177400                  MOV     #227,R0         ;;LOAD R0 WITH TEST NUMBER
5128  015056  000257                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5129  015060  000267                          MOV     #377,R4         ;RESULT S  /  B = 377
5130                                          MOV     #177400,R3      ;INITIAL DEST OP = 177400
5131  015062  000303                          CCC                     ;CLEAR FLAGS
5132                                          267                     ;MAKE N:C = 0111
5133  015064  100003
5134  015066  001402          2$:     SWAB    R3                      ;TEST THE SWAB
5135  015070  102401
5136  015072  103001                  BPL     3$
5137                                  BEQ     3$                      ;DID SWAB MAKE N:C = 1000
5138  015074  104002                  BVS     3$
5139                                  BCC     4$
5140  015076  020403
5141  015100  001401          3$:     ERROR   2                       ;SWAB FAILED TO ALTER CODES PROPERLY
5142
5143  015102  104002          4$:     CMP     R4,R3                   ;DID SWAB DELIVER CORRECT RESULT?
5144                                  BEQ     TST230                  ;;BR IF OK
5145
5146                          5$:     ERROR   2                       ;SWAB DELIVERED WRONG RESULT
5147
5148  015104                  ;;*****************************************************************
5149  015104  000004          ;*TEST 230       SWAB MODE 1 AND 2 TEST WITH POS. RESULT
5150  015106  012700  000230  ;;*****************************************************************
5151  015112  013701  015136  TST230:
5152  015116  012704  177400          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
5153  015122  012702  063312          MOV     #230,R0                 ;;LOAD R0 WITH TEST NUMBER
5154  015126  012712  000377          MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
5155  015132  000257                  MOV     #177400,R4              ;RESULT S  /  B = 177400
5156  015134  000273                  MOV     #MBUF0,R2               ;R2 POINTS TO DEST OP
5157                                  MOV     #377,(R2)               ;SET UP DEST OP = 377
5158  015136  000312                  CCC                             ;CLEAR FLAGS
5159                                  273                             ;MAKE N:C = 1011
5160  015140  100403
5161  015142  001002          2$:     SWAB    (R2)                    ;TEST THE SWAB - DM1
5162  015144  102401
5163  015146  103001                  BMI     3$
5164                                  BNE     3$                      ;N:C = 0100
5165  015150  104001                  BVS     3$
5166                                  BCC     4$
5167  015152  020412
5168  015154  001401          3$:     ERROR   1                       ;SWAB FAILED TO ALTER CODES PROPERLY
5169
5170  015156  104001          4$:     CMP     R4,(R2)                 ;CORRECT RESULT?
5171                                  BEQ     5$                      ;BR IF OK
5172  015160  013701  015200
5173  015164  012702  063312          ERROR   1                       ;SWAB DELIVERED WRONG RESULT
5174  015170  012712  000377
5175  015174  000257          5$:     MOV     @#20$,R1                ;LOAD R1 WITH TEST INSTR. WORD
5176  015176  000273                  MOV     #MBUF0,R2               ;R2 POINTS TO DEST OP
                                      MOV     #377,(R2)               ;[DEST] = 000377
                                      CCC                             ;CLEAR FLAGS
                                      273                             ;MAKE N:C = 1011
```

```
5177
5178  015200  000322                20$:    SWAB    (R2)+           ;TEST THE SWAB - DM2
5179
5180  015202  100403                        BMI     7$              ;N:C = 0100
5181  015204  001002                        BNE     7$
5182  015206  102401                        BVS     7$
5183  015210  103001                        BCC     6$
5184
5185  015212  104001                7$:     ERROR   1               ;SWAB FAILED TO SET CODES PROPERLY
5186
5187  015214  020437  063312        6$:     CMP     R4,@#MBUF0      ;CORRECT RESULT ?
5188  015220  001401                        BEQ     8$              ;BR IF YES
5189
5190  015222  104001                9$:     ERROR   1               ;SWAB DELIVERED THE WRONG RESULT
5191
5192  015224  020227  063314        8$:     CMP     R2,#MBUF0+2     ;DID AUTO INCREMENT OCCUR ?
5193  015230  001401                        BEQ     TST231          ;;BR IF YES
5194
5195  015232  104001                        ERROR   1               ;SWAB FAILED TO AUTO INC REG.
5196
5197                                ;;*******************************************************************
5198                                ;*TEST 231       SWAB MODE 1 TEST WITH NEG. RESULT
5199                                ;;*******************************************************************
5200  015234                        TST231:
5201  015234  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5202  015236  012700  000231                MOV     #231,R0         ;;LOAD R0 WITH TEST NUMBER
5203  015242  013701  015266                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5204  015246  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
5205  015252  012702  063312                MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5206  015256  012712  177400                MOV     #177400,(R2)    ;SET UP DEST. OP = 177400
5207  015262  000257                        CCC                     ;CLEAR FLAGS
5208  015264  000267                        267                     ;MAKE N:C = 0111
5209
5210  015266  000312                2$:     SWAB    (R2)            ;TEST THE SWAB
5211
5212  015270  100003                        BPL     3$
5213  015272  001402                        BEQ     3$              ;N:C = 1000?
5214  015274  102401                        BVS     3$
5215  015276  103001                        BCC     4$
5216
5217  015300  104001                3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5218
5219  015302  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5220  015304  001401                        BEQ     TST232          ;;BR IF YES
5221
5222  015306  104001                5$:     ERROR   1               ;SWAB DELIVERED WRONG RESULT
5223
5224                                ;;*******************************************************************
5225                                ;*TEST 232       NEG MODE 0 TEST : [DEST] = 0
5226                                ;;*******************************************************************
5227  015310                        TST232:
5228  015310  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5229  015312  012700  000232                MOV     #232,R0         ;;LOAD R0 WITH TEST NUMBER
5230  015316  013701  015332                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5231  015322  005004                        CLR     R4              ;RESULT S / B = 0
5232  015324  005003                        CLR     R3              ;INITIAL [DEST] = 0
```

```
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;MAKE N:C = 1011

                            2$:         NEG     R3              ;TEST THE NEG

                                        BMI     3$
                                        BNE     3$              ;N:C = 0100 ONLY ''Z'' SET?
                                        BVS     3$
                                        BCC     4$

                            3$:         ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY

                            4$:         CMP     R3,R4           ;WAS RESULT = 0
                                        BEQ     TST233          ;;BR IF YES

                            5$:         ERROR   2               ;NEG DELIVERED WRONG RESULT

                            ;;**********************************************************************
                            ;*TEST 233      NEG MODE 0 TEST : [DEST] LT 0
                            ;;**********************************************************************
015354                      TST233:
015354  000004                      SCOPE                       ;CALL THE SCOPE LOOP UTILITY
015356  012700  000233              MOV     #233,R0             ;;LOAD R0 WITH TEST NUMBER
015362  013701  015402              MOV     @#2$,R1             ;LOAD R1 WITH TEST INSTRUCTION WORD
015366  012704  000002              MOV     #2,R4               ;RESULT S / B = 2
015372  012703  177776              MOV     #-2,R3              ;INITIAL [DEST] = 177776
015376  000257                      CCC                         ;CLEAR FLAGS
015400  000276                      276                         ;MAKE N:C = 1110

015402  005403              2$:     NEG     R3                  ;TEST THE NEG

015404  100403                      BMI     3$
015406  001402                      BEQ     3$                  ;N:C = 0001?
015410  102401                      BVS     3$
015412  103401                      BCS     4$

015414  104002              3$:     ERROR   2                   ;NEG FAILED TO ALTER CODES PROPERLY

015416  020304              4$:     CMP     R3,R4               ;RESULT = 2?
015420  001401                      BEQ     TST234              ;;BR IF YES

015422  104002              5$:     ERROR   2                   ;NEG DELIVERED WRONG RESULT

                            ;;**********************************************************************
                            ;*TEST 234      NEG MODE 0 TEST : [DEST] = 100000 (8)
                            ;;**********************************************************************
015424                      TST234:
015424  000004                      SCOPE                       ;CALL THE SCOPE LOOP UTILITY
015426  012700  000234              MOV     #234,R0             ;;LOAD R0 WITH TEST NUMBER
015432  013701  015450              MOV     @#2$,R1             ;LOAD R1 WITH TEST INSTRUCTION WORD
015436  012704  100000              MOV     #100000,R4          ;RESULT S / B = 100000
015442  010403                      MOV     R4,R3               ;INITIAL [DEST] = 100000
015444  000257                      CCC                         ;CLEAR FLAGS
015446  000264                      SEZ                         ;MAKE N:C = 01000

015450  005403              2$:     NEG     R3                  ;TEST THE NEG
```

```
5289
5290  015452  100003                  BPL     3$
5291  015454  001402                  BEQ     3$              ;N:C = 1011?
5292  015456  102001                  BVC     3$
5293  015460  103401                  BCS     4$
5294
5295  015462  104002          3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5296
5297  015464  020304          4$:     CMP     R3,R4           ;RESULT STILL 100000?
5298  015466  001401                  BEQ     TST235          ;;BR IF YES
5299
5300  015470  104002          5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5301
5302                          ;;**************************************************************
5303                          ;*TEST 235       NEG MODE 1 TEST : [DEST] = 0
5304                          ;;**************************************************************
5305  015472                  TST235:
5306  015472  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5307  015474  012700  000235          MOV     #235,R0         ;;LOAD R0 WITH TEST NUMBER
5308  015500  013701  015520          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5309  015504  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5310  015510  005004                  CLR     R4              ;RESULT S / B = 0
5311  015512  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
5312  015514  000257                  CCC                     ;CLEAR FLAGS
5313  015516  000273                  273                     ;MAKE N:C = 1011
5314
5315  015520  005412          2$:     NEG     (R2)            ;TEST THE NEG
5316
5317  015522  100403                  BMI     3$
5318  015524  001002                  BNE     3$              ;N:C = 0100?
5319  015526  102401                  BVS     3$
5320  015530  103001                  BCC     4$
5321
5322  015532  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5323
5324  015534  021204          4$:     CMP     (R2),R4         ;RESULT = 0?
5325  015536  001401                  BEQ     TST236          ;;BR IF YES
5326
5327  015540  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5328
5329                          ;;**************************************************************
5330                          ;*TEST 236       NEG MODE 1  TEST : [DEST] GT 0
5331                          ;;**************************************************************
5332  015542                  TST236:
5333  015542  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5334  015544  012700  000236          MOV     #236,R0         ;;LOAD R0 WITH TEST NUMBER
5335  015550  013701  015574          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5336  015554  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5337  015560  012704  177776          MOV     #-2,R4          ;RESULT S / B = 177776
5338  015564  012712  000002          MOV     #2,(R2)         ;INITIAL [DEST] = 2
5339  015570  000257                  CCC                     ;CLEAR FLAGS
5340  015572  000266                  266                     ;MAKE N:C = 0110
5341
5342  015574  005412          2$:     NEG     (2)             ;TEST THE NEG
5343
5344  015576  100003                  BPL     3$
```

```
5345  015600  001402                  BEQ     3$              ;N:C = 1001?
5346  015602  102401                  BVS     3$
5347  015604  103401                  BCS     4$
5348
5349  015606  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5350
5351  015610  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT?
5352  015612  001401                  BEQ     TST237          ;;BR IF YES
5353
5354  015614  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5355
5356                          ;;**********************************************************************
5357                          ;*TEST 237       NEG MODE 1 TEST : [DEST] LT 0
5358                          ;;**********************************************************************
5359  015616                  TST237:
5360  015616  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5361  015620  012700  000237          MOV     #237,R0         ;;LOAD R0 WITH TEST NUMBER
5362  015624  013701  015650          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5363  015630  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5364  015634  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
5365  015640  012712  177776          MOV     #-2,(R2)        ;INITIAL [DEST] = 177776
5366  015644  000257                  CCC                     ;CLEAR FLAGS
5367  015646  000276                  276                     ;MAKE N:C = 1110
5368
5369  015650  005412          2$:     NEG     (R2)            ;TEST THE NEG
5370
5371  015652  100403                  BMI     3$
5372  015654  001402                  BEQ     3$              ;N:C = 0001?
5373  015656  102401                  BVS     3$
5374  015660  103401                  BCS     4$
5375
5376  015662  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5377
5378  015664  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT = 2?
5379  015666  001401                  BEQ     TST240          ;;BR IF YES
5380
5381  015670  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5382
5383                          ;;**********************************************************************
5384                          ;*TEST 240       NEG MODE 1 TEST: [DEST] = 100000 (8)
5385                          ;;**********************************************************************
5386  015672                  TST240:
5387  015672  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5388  015674  012700  000240          MOV     #240,R0         ;;LOAD R0 WITH TEST NUMBER
5389  015700  013701  015722          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5390  015704  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5391  015710  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
5392  015714  010412                  MOV     R4,(R2)         ;INITIAL [DEST] = 100000
5393  015716  000257                  CCC                     ;CLEAR FLAGS
5394  015720  000264                  SEZ                     ;MAKE N:Z = 0100
5395
5396  015722  005412          2$:     NEG     (R2)            ;TEST THE NEG
5397
5398  015724  100003                  BPL     3$
5399  015726  001402                  BEQ     3$              ;N:C = 1011?
5400  015730  102001                  BVC     3$
```

```
5401  015732  103401                  BCS     4$
5402
5403  015734  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5404
5405  015736  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT = 100000?
5406  015740  001401                  BEQ     TST241          ;;BR IF YES
5407
5408  015742  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5409
5410                          ;;*********************************************************************
5411                          ;*TEST 241       ROR TEST - DM0 - N:C = 1110
5412                          ;;*********************************************************************
5413  015744                  TST241:
5414  015744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5415  015746  012700  000241          MOV     #241,R0         ;;LOAD R0 WITH TEST NUMBER
5416  015752  013701  015772          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5417  015756  012704  052525          MOV     #52525,R4       ;RESULT S / B = 52525
5418  015762  012703  125252          MOV     #125252,R3      ;[DEST] = 125252
5419  015766  000257                  CCC                     ;CLEAR FLAGS
5420  015770  000276                  276                     ;N:C = 1111
5421
5422  015772  006003          2$:     ROR     R3              ;TEST THE ROR
5423
5424  015774  100403                  BMI     3$              ;N:C = 0000 ?
5425  015776  001402                  BEQ     3$
5426  016000  102401                  BVS     3$
5427  016002  103001                  BCC     4$
5428
5429  016004  104002          3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5430
5431  016006  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
5432  016010  001401                  BEQ     TST242          ;;BR IF YES
5433
5434  016012  104002          5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5435
5436                          ;;*********************************************************************
5437                          ;*TEST 242       ROR TEST - DM0 - N:C = 1000
5438                          ;;*********************************************************************
5439  016014                  TST242:
5440  016014  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5441  016016  012700  000242          MOV     #242,R0         ;;LOAD R0 WITH TEST NUMBER
5442  016022  013701  016040          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5443  016026  005004                  CLR     R4              ;RESULT S / B = 000000
5444  016030  012703  000001          MOV     #1,R3           ;[DEST] = 1
5445  016034  000257                  CCC                     ;CLEAR FLAGS
5446  016036  000270                  SEN                     ;N:C = 1000
5447
5448  016040  006003          2$:     ROR     R3              ;TEST THE ROR
5449
5450  016042  100403                  BMI     3$              ;N:C = 0111 ?
5451  016044  001002                  BNE     3$
5452  016046  102001                  BVC     3$
5453  016050  103401                  BCS     4$
5454
5455  016052  104002          3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5456
```

```
5457  016054  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5458  016056  001401                          BEQ     TST243          ;;BR IF YES
5459
5460  016060  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5461
5462                                  ;;**********************************************************************
5463                                  ;*TEST 243       ROR TEST - DM0 - N:C = 0111
5464                                  ;;**********************************************************************
5465  016062                          TST243:
5466  016062  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5467  016064  012700  000243                  MOV     #243,R0         ;;LOAD R0 WITH TEST NUMBER
5468  016070  013701  016110                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5469  016074  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
5470  016100  012703  052525                  MOV     #52525,R3       ;[DEST] = 052525
5471  016104  000257                          CCC                     ;CLEAR FLAGS
5472  016106  000267                          267                     ;N:C = 0111
5473
5474  016110  006003                  2$:     ROR     R3              ;TEST THE ROR
5475
5476  016112  100003                          BPL     3$              ;N:C = 1001 ?
5477  016114  001402                          BEQ     3$
5478  016116  102401                          BVS     3$
5479  016120  103401                          BCS     4$
5480
5481  016122  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5482
5483  016124  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5484  016126  001401                          BEQ     TST244          ;;BR IF YES
5485
5486  016130  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5487
5488                                  ;;**********************************************************************
5489                                  ;*TEST 244       ASR TEST - DM0 - N:C = 1000
5490                                  ;;**********************************************************************
5491  016132                          TST244:
5492  016132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5493  016134  012700  000244                  MOV     #244,R0         ;;LOAD R0 WITH TEST NUMBER
5494  016140  013701  016156                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5495  016144  005004                          CLR     R4              ;RESULT S / B = 000000
5496  016146  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5497  016152  000257                          CCC                     ;CLEAR FLAGS
5498  016154  000270                          SEN                     ;N:C = 1000
5499
5500  016156  006003                  2$:     ROR     R3              ;TEST THE ROR
5501
5502  016160  100403                          BMI     3$              ;N:C = 0111 ?
5503  016162  001002                          BNE     3$
5504  016164  102001                          BVC     3$
5505  016166  103401                          BCS     4$
5506
5507  016170  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5508
5509  016172  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5510  016174  001401                          BEQ     TST245          ;;BR IF YES
5511
5512  016176  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 245       ASR TEST - DM0 - N:C = 0101
                                ;;*****************************************************************
016200                          TST245:
016200  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
016202  012700  000245                  MOV     #245,R0         ;;LOAD R0 WITH TEST NUMBER
016206  013701  016226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
016212  012704  152525                  MOV     #152525,R4      ;RESULT S / B = 152525
016216  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
016222  000257                          CCC                     ;CLEAR FLAGS
016224  000265                          265                     ;N:C = 0101
                                
016226  006003                  2$:     ROR     R3              ;TEST THE ROR
                                
016230  100003                          BPL     3$              ;N:C = 1010 ?
016232  001402                          BEQ     3$
016234  102001                          BVC     3$
016236  103001                          BCC     4$
                                
016240  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
                                
016242  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
016244  001401                          BEQ     TST246          ;;BR IF YES
                                
016246  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 246       ASR TEST - DM0 - N:C = 1100
                                ;;*****************************************************************
016250                          TST246:
016250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
016252  012700  000246                  MOV     #246,R0         ;;LOAD R0 WITH TEST NUMBER
016256  013701  016276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
016262  012704  025252                  MOV     #25252,R4       ;RESULT S / B = 25252
016266  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
016272  000257                          CCC                     ;CLEAR FLAGS
016274  000274                          274                     ;N:C = 1100
                                
016276  006003                  2$:     ROR     R3              ;TEST THE ROR
                                
016300  100403                          BMI     3$              ;N:C = 0011 ?
016302  001402                          BEQ     3$
016304  102001                          BVC     3$
016306  103401                          BCS     4$
                                
016310  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
                                
016312  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
016314  001401                          BEQ     TST247          ;;BR IF YES
                                
016316  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 247       ROR TEST - DM1 - N:C = 1110
                                ;;*****************************************************************
```

```
5569  016320
5570  016320  000004
5571  016322  012700  000247          TST247:
5572  016326  013701  016352                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5573  016332  012702  063312                  MOV     #247,R0         ;;LOAD R0 WITH TEST NUMBER
5574  016336  012704  052525                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5575  016342  012712  125252                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5576  016346  000257                          MOV     #52525,R4       ;RESULT S / B = 52525
5577  016350  000276                          MOV     #125252,(R2)    ;[DEST] = 125252
5578                                          CCC                     ;CLEAR FLAGS
5579  016352  006012                  2$:     276                     ;N:C = 1110
5580                                          ROR     (R2)            ;TEST THE ROR
5581  016354  100403
5582  016356  001402                          BMI     3$              ;N:C = 0000 ?
5583  016360  102401                          BEQ     3$
5584  016362  103001                          BVS     3$
5585                                          BCC     4$
5586  016364  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5587
5588  016366  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5589  016370  001402                          BEQ     TST250          ;;BR IF YES
5590  016372  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5591  016374  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5592
5593                                  ;;**********************************************************************
5594                                  ;*TEST 250       ROR TEST - DM1 - N:C = 1000
5595                                  ;;**********************************************************************
5596  016376                          TST250:
5597  016376  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5598  016400  012700  000250                  MOV     #250,R0         ;;LOAD R0 WITH TEST NUMBER
5599  016404  013701  016426                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5600  016410  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5601  016414  005004                          CLR     R4              ;RESULT S / B = 000000
5602  016416  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
5603  016422  000257                          CCC                     ;CLEAR FLAGS
5604  016424  000270                          SEN                     ;N:C = 1000
5605
5606  016426  006012                  2$:     ROR     (R2)            ;TEST THE ROR
5607
5608  016430  100403                          BMI     3$              ;N:C = 0111 ?
5609  016432  001002                          BNE     3$
5610  016434  102001                          BVC     3$
5611  016436  103401                          BCS     4$
5612
5613  016440  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5614
5615  016442  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5616  016444  001402                          BEQ     TST251          ;;BR IF YES
5617
5618  016446  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5619  016450  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5620
5621                                  ;;**********************************************************************
5622                                  ;*TEST 251       ROR TEST - DM1 - N:C = 0111
5623                                  ;;**********************************************************************
5624  016452                          TST251:
```

```
5625  016452  000004
5626  016454  012700  000251          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5627  016460  013701  016504          MOV     #251,R0         ;;LOAD R0 WITH TEST NUMBER
5628  016464  012702  063312          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5629  016470  012704  125252          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5630  016474  012712  052525          MOV     #125252,R4      ;RESULT S / B = 125252
5631  016500  000257                  MOV     #52525,(R2)     ;[DEST] = 52525
5632  016502  000267                  CCC                     ;CLEAR FLAGS
5633                                   267                     ;N:C = 0111
5634  016504  006012          2$:     ROR     (R2)            ;TEST THE ROR
5635
5636  016506  100003                  BPL     3$              ;N:C = 1001 ?
5637  016510  001402                  BEQ     3$
5638  016512  102401                  BVS     3$
5639  016514  103401                  BCS     4$
5640
5641  016516  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5642
5643  016520  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5644  016522  001402                  BEQ     TST252          ;;BR IF YES
5645
5646  016524  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5647  016526  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5648
5649                          ;;**********************************************************************
5650                          ;*TEST 252       ASR TEST - DM1 - N:C = 1000
5651                          ;;**********************************************************************
5652  016530                  TST252:
5653  016530  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5654  016532  012700  000252          MOV     #252,R0         ;;LOAD R0 WITH TEST NUMBER
5655  016536  013701  016560          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5656  016542  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5657  016546  005004                  CLR     R4              ;RESULT S  /  B = 000000
5658  016550  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
5659  016554  000257                  CCC                     ;CLEAR FLAGS
5660  016556  000270                  SEN                     ;N:C = 1000
5661
5662  016560  006012          2$:     ROR     (R2)            ;TEST THE ROR
5663
5664  016562  100403                  BMI     3$              ;N:C = 0111 ?
5665  016564  001002                  BNE     3$
5666  016566  102001                  BVC     3$
5667  016570  103401                  BCS     4$
5668
5669  016572  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5670
5671  016574  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5672  016576  001402                  BEQ     TST253          ;;BR IF YES
5673
5674  016600  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5675  016602  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5676
5677                          ;;**********************************************************************
5678                          ;*TEST 253       ASR TEST - DM1 - N:C = 1100
5679                          ;;**********************************************************************
5680  016604                  TST253:
```

```
5681  016604  000004
5682  016606  012700  000253          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5683  016612  013701  016636          MOV     #253,R0         ;;LOAD R0 WITH TEST NUMBER
5684  016616  012702  063312          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5685  016622  012704  025252          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5686  016626  012712  052525          MOV     #25252,R4       ;RESULT S  /  B = 25252
5687  016632  000257                  MOV     #52525,(R2)     ;[DEST] = 52525
5688  016634  000274                  CCC                     ;CLEAR FLAGS
5689                                  274                     ;N:C = 1100
5690  016636  006012          2$:     ROR     (R2)            ;TEST THE ROR
5691
5692  016640  100403                  BMI     3$              ;N:C = 0011 ?
5693  016642  001402                  BEQ     3$
5694  016644  102001                  BVC     3$
5695  016646  103401                  BCS     4$
5696
5697  016650  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5698
5699  016652  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5700  016654  001402                  BEQ     TST254          ;;BR IF YES
5701
5702  016656  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5703  016660  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5704
5705                          ;;**********************************************************************
5706                          ;*TEST 254       ASR TEST - DM1 - N:C = 0101
5707                          ;;**********************************************************************
5708  016662                  TST254:
5709  016662  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5710  016664  012700  000254          MOV     #254,R0         ;;LOAD R0 WITH TEST NUMBER
5711  016670  013701  016714          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5712  016674  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5713  016700  012704  152525          MOV     #152525,R4      ;RESULT S / B = 152525
5714  016704  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
5715  016710  000257                  CCC                     ;CLEAR FLAGS
5716  016712  000265                  265                     ;N:C = 0101
5717
5718  016714  006012          2$:     ROR     (R2)            ;TEST THE ROR
5719
5720  016716  100003                  BPL     3$              ;N:C = 1010 ?
5721  016720  001402                  BEQ     3$
5722  016722  102001                  BVC     3$
5723  016724  103001                  BCC     4$
5724
5725  016726  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5726
5727  016730  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5728  016732  001402                  BEQ     TST255          ;;BR IF YES
5729
5730  016734  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5731  016736  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5732
5733                          ;;**********************************************************************
5734                          ;*TEST 255       RORB TEST - DM2 - EVEN ADDRESS
5735                          ;;**********************************************************************
5736  016740                  TST255:
```

```
5737  016740  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5738  016742  012700  000255            MOV     #255,R0         ;;LOAD R0 WITH TEST NUMBER
5739  016746  013701  016772            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5740  016752  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5741  016756  012704  000177            MOV     #177,R4         ;RESULT S / B = 177
5742  016762  010203                    MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5743  016764  012712  000377            MOV     #377,(R2)       ;[DEST] = 377
5744  016770  000257                    CCC                     ;SCOPE SYNC ''C'' = 0
5745
5746  016772  106023            2$:     RORB    (R3)+           ;TEST THE RORB
5747
5748  016774  103401                    BCS     4$              ;BR IF ROR SET ''C''
5749
5750  016776  104001            3$:     ERROR   1               ;ROR FAILED TO SET ''C''
5751
5752  017000  022703  063313    4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5753  017004  001401                    BEQ     6$              ;BR IF YES
5754
5755  017006  104005            5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5756
5757  017010  020412            6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5758  017012  001402                    BEQ     TST256          ;;BR IF YES
5759
5760  017014  011203                    MOV     (R2),R3         ;GET THE WAS DATA
5761  017016  104001            7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5762
5763                            ;;**************************************************************
5764                            ;*TEST 256      RORB TEST - DM1 - EVEN ADDRESS
5765                            ;;**************************************************************
5766  017020                    TST256:
5767  017020  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5768  017022  012700  000256            MOV     #256,R0         ;;LOAD R0 WITH TEST NUMBER
5769  017026  013701  017054            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5770  017032  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5771  017036  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
5772  017042  010203                    MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5773  017044  012712  000376            MOV     #376,(R2)       ;[DEST] = 376
5774  017050  000257                    CCC                     ;CLEAR FLAGS
5775  017052  000261                    SEC                     ;SCOPE SYNC - SET ''C''
5776
5777  017054  106013            2$:     RORB    (R3)            ;TEST THE RORB
5778
5779  017056  103001                    BCC     4$              ;BR IF ''C'' CLR - IT SHOULD BE
5780
5781  017060  104001            3$:     ERROR   1               ;RORB FAILED TO CLR ''C''
5782
5783  017062  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5784  017064  001402                    BEQ     TST257          ;;BR IF YES
5785
5786  017066  011203                    MOV     (R2),R3         ;GET THE WAS DATA
5787  017070  104001            5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5788
5789                            ;;**************************************************************
5790                            ;*TEST 257      RORB TEST - DM2 - ODD ADDRESS
5791                            ;;**************************************************************
5792  017072                    TST257:
```

```
5793  017072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5794  017074  012700  000257                  MOV     #257,R0         ;;LOAD R0 WITH TEST NUMBER
5795  017100  013701  017142                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5796                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT5
5797  017104  032737  000040  063234          BIT     #BIT5,@#BPTLOC  ;BREAKPOINT HALT SET ??
5798  017112  001401                          BEQ     .+4             ;BR IF NOT
5799  017114  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
5800  017116  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5801  017122  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
5802  017126  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5803  017132  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5804  017134  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
5805  017140  000257                          CCC                     ;SCOPE SYNC - "C" =0
5806
5807  017142  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5808
5809  017144  103401                          BCS     4$              ;BR IF "C" IS SET - IT SHOULD BE
5810
5811  017146  104001                  3$:     ERROR   1               ;RORB FAILED TO SET "C"
5812
5813  017150  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5814  017154  001401                          BEQ     6$              ;BR IF YES
5815
5816  017156  104005                  5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5817
5818  017160  020415                  6$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5819  017162  001402                          BEQ     TST260          ;;BR IF YES
5820
5821  017164  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5822  017166  104001                  7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5823
5824                                  ;;*******************************************************************
5825                                  ;*TEST 260       RORB TEST - DM1 - ODD ADDRESS
5826                                  ;;*******************************************************************
5827  017170                          TST260:
5828  017170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5829  017172  012700  000260                  MOV     #260,R0         ;;LOAD R0 WITH TEST NUMBER
5830  017176  013701  017226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5831  017202  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5832  017206  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
5833  017212  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5834  017216  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5835  017220  012715  177377                  MOV     #177377,(R5)    ;[DEST] = 177377
5836  017224  000261                          SEC                     ;SCOPE SYNC - SET "C"
5837
5838  017226  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5839
5840  017230  103001                          BCC     4$              ;BR IF "C" CLEAR - IT SHOULD BE
5841
5842  017232  104001                  3$:     ERROR   1               ;RORB FAILED TO CLEAR "C"
5843
5844  017234  020415                  4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5845  017236  001402                          BEQ     TST261          ;;BR IF YES
5846
5847  017240  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5848  017242  104001                  5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
```

```
5849
5850                            ;;*****************************************************************
5851                            ;*TEST 261       ASRB TEST - DM2 - ODD ADDRESS
5852                            ;;*****************************************************************
5853  017244                    TST261:
5854  017244  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5855  017246  012700  000261           MOV     #261,R0         ;;LOAD R0 WITH TEST NUMBER
5856  017252  013701  017302           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5857  017256  012702  063313           MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5858  017262  012704  000377           MOV     #377,R4         ;RESULT S / B = 377
5859  017266  012705  063312           MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5860  017272  010203                   MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5861  017274  012715  000777           MOV     #777,(R5)       ;[DEST] = 777
5862  017300  000257                   CCC                     ;SCOPE SYNC ''C'' = 0
5863
5864  017302  106223            2$:    ASRB    (R3)+           ;TEST THE ASRB
5865
5866  017304  103401                   BCS     4$              ;BR IF CARRY SET - IT SHOULD BE
5867
5868  017306  104001            3$:    ERROR   1               ;ASRB FAILED TO SET THE CARRY
5869
5870  017310  022703  063314    4$:    CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5871  017314  001401                   BEQ     6$              ;BR IF YES
5872
5873  017316  104005            5$:    ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5874
5875  017320  020415            6$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5876  017322  001402                   BEQ     TST262          ;;BR IF YES
5877
5878  017324  011503                   MOV     (R5),R3         ;GET THE WAS DATA
5879  017326  104001            7$:    ERROR   1               ;ASRB DELIVERED WRONG RESULT
5880
5881                            ;;*****************************************************************
5882                            ;*TEST 262       ASRB TEST - DM1 - ODD ADDRESS
5883                            ;;*****************************************************************
5884  017330                    TST262:
5885  017330  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5886  017332  012700  000262           MOV     #262,R0         ;;LOAD R0 WITH TEST NUMBER
5887  017336  013701  017366           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5888  017342  012702  063313           MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5889  017346  012704  140377           MOV     #140377,R4      ;RESULT S / B = 140377
5890  017352  012705  063312           MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5891  017356  010203                   MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5892  017360  012715  100377           MOV     #100377,(R5)    ;[DEST] = 100377
5893  017364  000261                   SEC                     ;SCOPE SYNC - ''C'' = 1
5894
5895  017366  106213            2$:    ASRB    (R3)            ;TEST THE ASRB
5896
5897  017370  103001                   BCC     4$              ;BR IF CARRY CLEAR - IT SHOULD BE
5898
5899  017372  104001            3$:    ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5900
5901  017374  020415            4$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5902  017376  001402                   BEQ     TST263          ;;BR IF YES
5903
5904  017400  011503                   MOV     (R5),R3         ;GET THE WAS DATA
```

```
5905  017402  104001                5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5906
5907                                ;;*****************************************************************
5908                                ;*TEST 263       ASRB TEST - DM2 - EVEN ADDRESS
5909                                ;;*****************************************************************
5910  017404                        TST263:
5911  017404  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5912  017406  012700  000263               MOV     #263,R0         ;;LOAD R0 WITH TEST NUMBER
5913  017412  013701  017436               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5914  017416  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5915  017422  012704  000077               MOV     #77,R4          ;RESULT S / B = 77
5916  017426  010203                       MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5917  017430  012712  000177               MOV     #177,(R2)       ;[DEST] = 177
5918  017434  000257                       CCC                     ;SCOPE SYNC - "C" = 0
5919
5920  017436  106223                2$:     ASRB    (R3)+           ;TEST THE ASRB
5921
5922  017440  103401                       BCS     4$              ;BR IF "C" = 1 - IT SHOULD BE
5923
5924  017442  104001                3$:     ERROR   1               ;ASRB FAILED TO SET "C"
5925
5926  017444  022703  063313        4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5927  017450  001401                       BEQ     6$              ;BR IF YES
5928
5929  017452  104005                5$:     ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5930
5931  017454  020412                6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5932  017456  001402                       BEQ     TST264          ;;BR IF YES
5933
5934  017460  011203                       MOV     (R2),R3         ;GET THE WAS DATA
5935  017462  104001                7$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5936
5937                                ;;*****************************************************************
5938                                ;*TEST 264       ASRB TEST - DM1 - EVEN ADDRESS
5939                                ;;*****************************************************************
5940  017464                        TST264:
5941  017464  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5942  017466  012700  000264               MOV     #264,R0         ;;LOAD R0 WITH TEST NUMBER
5943  017472  013701  017516               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5944  017476  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5945  017502  012704  000303               MOV     #303,R4         ;RESULT S / B = 303
5946  017506  010203                       MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5947  017510  012712  000206               MOV     #206,(R2)       ;[DEST] = 206
5948  017514  000261                       SEC                     ;SCOPE SYNC - "C" = 1
5949
5950  017516  106213                2$:     ASRB    (R3)            ;TEST THE CLRASRB
5951
5952  017520  103001                       BCC     4$              ;BR IF CARRY CLEAR - ITSHOULD BE
5953
5954  017522  104001                3$:     ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5955
5956  017524  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5957  017526  001402                       BEQ     TST265          ;;BR IF YES
5958
5959  017530  011203                       MOV     (R2),R3         ;GET THE WAS DATA
5960  017532  104001                5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 265       TST DM0 TEST - N:C = 1011
                                ;;*****************************************************************
017534                          TST265:
017534  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017536  012700  000265                  MOV     #265,R0         ;;LOAD R0 WITH TEST NUMBER
017542  013701  017556                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017546  005004                          CLR     R4              ;RESULT S / B = 000000
017550  005003                          CLR     R3              ;[DEST] = 000000
017552  000257                          CCC                     ;CLEAR CODES
017554  000273                          273                     ;N:C=1011

017556  005703                  2$:     TST     R3              ;TEST THE TST

017560  100403                          BMI     3$              ;N:C = 0100 ?
017562  001002                          BNE     3$
017564  102401                          BVS     3$
017566  103001                          BCC     4$

017570  104002                  3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY

017572  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017574  001401                          BEQ     TST266          ;;BR IF YES

017576  104002                  5$:     ERROR   2               ;TST ALTERED THE [DEST]

                                ;;*****************************************************************
                                ;*TEST 266       TST DM0 TEST - N:C = 0100
                                ;;*****************************************************************
017600                          TST266:
017600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017602  012700  000266                  MOV     #266,R0         ;;LOAD R0 WITH TEST NUMBER
017606  013701  017624                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017612  005004                          CLR     R4
017614  005104                          COM     R4              ;RESULT S / B = 177777
017616  010403                          MOV     R4,R3           ;[DEST] = 177777
017620  000257                          CCC                     ;CLEAR CODES
017622  000264                          264                     ;N:C=0100

017624  005703                  2$:     TST     R3              ;TEST THE TST

017626  100003                          BPL     3$              ;N:C = 1000 ?
017630  001402                          BEQ     3$
017632  102401                          BVS     3$
017634  103001                          BCC     4$

017636  104002                  3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY

017640  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017642  001401                          BEQ     TST267          ;;BR IF YES

017644  104002                  5$:     ERROR   2               ;TST ALTERED THE [DEST]

                                ;;*****************************************************************
                                ;*TEST 267       CLR DM0 TEST - N:C = 1011
```

```
6017
6018  017646
6019  017646  000004
6020  017650  012700  000267          ;;*****************************************************************
6021  017654  013701  017672          TST267:
6022  017660  005004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6023  017662  012703  177777                  MOV     #267,R0         ;;LOAD R0 WITH TEST NUMBER
6024  017666  000257                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6025  017670  000273                          CLR     R4              ;RESULT S / B = 000000
6026                                          MOV     #-1,R3          ;[DEST] = 177777
6027  017672  005003                          CCC                     ;CLEAR CODES
6028                                          273                     ;N:C = 1011
6029  017674  100403                  2$:     CLR     R3              ;TEST THE CLR
6030  017676  001002
6031  017700  102401                          BMI     3$              ;N:C = 0100 ?
6032  017702  103001                          BNE     3$
6033                                          BVS     3$
6034  017704  104002                          BCC     4$
6035
6036  017706  020403                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6037  017710  001401
6038                                  4$:     CMP     R4,R3           ;RESULT OK ?
6039  017712  104002                          BEQ     TST270          ;;BR IF YES
6040
6041                                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6042
6043                                  ;;*****************************************************************
6044  017714                          ;*TEST 270       CLR DM0 TEST - N:C = 0000
6045  017714  000004                  ;;*****************************************************************
6046  017716  012700  000270          TST270:
6047  017722  013701  017736                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6048  017726  005004                          MOV     #270,R0         ;;LOAD R0 WITH TEST NUMBER
6049  017730  012703  177777                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6050  017734  000257                          CLR     R4              ;RESULT S / B = 000000
6051                                          MOV     #-1,R3          ;[DEST] = 177777
6052  017736  005003                          CCC                     ;CLEAR CODES
6053                                  2$:     CLR     R3              ;TEST THE CLR
6054  017740  100403
6055  017742  001002                          BMI     3$              ;N:C = 0100 ?
6056  017744  102401                          BNE     3$
6057  017746  103001                          BVS     3$
6058                                          BCC     4$
6059  017750  104002
6060                                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
6061  017752  020403
6062  017754  001401                  4$:     CMP     R4,R3           ;RESULT OK ?
6063                                          BEQ     TST271          ;;BR IF YES
6064  017756  104002
6065                                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
6066
6067                                  ;;*****************************************************************
6068                                  ;*TEST 271       COM DM0 TEST - N:C = 0110
6069  017760                          ;;*****************************************************************
6070  017760  000004                  TST271:
6071  017762  012700  000271                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6072  017766  013701  020006                  MOV     #271,R0         ;;LOAD R0 WITH TEST NUMBER
                                              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
6073  017772  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6074  017776  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
6075  020002  000257                  CCC                     ;CLEAR CODES
6076  020004  000266                  266                     ;N:C = 0110
6077
6078  020006  005103          2$:     COM     R3              ;TEST THE COM
6079
6080  020010  100003                  BPL     3$              ;N:C = 1001 ?
6081  020012  001402                  BEQ     3$
6082  020014  102401                  BVS     3$
6083  020016  103401                  BCS     4$
6084
6085  020020  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6086
6087  020022  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6088  020024  001401                  BEQ     TST272          ;;BR IF YES
6089
6090  020026  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6091
6092                          ;;**************************************************************
6093                          ;*TEST 272       COM DM0 TEST - N:C = 1001
6094                          ;;**************************************************************
6095  020030                  TST272:
6096  020030  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6097  020032  012700  000272          MOV     #272,R0         ;;LOAD R0 WITH TEST NUMBER
6098  020036  013701  020054          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6099  020042  005004                  CLR     R4              ;RESULT S / B = 000000
6100  020044  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6101  020050  000257                  CCC                     ;CLEAR CODES
6102  020052  000271                  271                     ;N:C = 1001
6103
6104  020054  005103          2$:     COM     R3              ;TEST THE COM
6105
6106  020056  100403                  BMI     3$              ;N:C = 0101 ?
6107  020060  001002                  BNE     3$
6108  020062  102401                  BVS     3$
6109  020064  103401                  BCS     4$
6110
6111  020066  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
6112
6113  020070  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6114  020072  001401                  BEQ     TST273          ;;BR IF YES
6115
6116  020074  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
6117
6118                          ;;**************************************************************
6119                          ;*TEST 273       INC DM0 TEST - N:C = 1011
6120                          ;;**************************************************************
6121  020076                  TST273:
6122  020076  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6123  020100  012700  000273          MOV     #273,R0         ;;LOAD R0 WITH TEST NUMBER
6124  020104  013701  020122          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6125  020110  005004                  CLR     R4              ;RESULT S / B = 000000
6126  020112  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
6127  020116  000257                  CCC                     ;CLEAR CODES
6128  020120  000273                  273                     ;N:C = 1011
```

```
6129
6130  020122  005203                  2$:     INC     R3              ;TEST THE INC
6131
6132  020124  100403                          BMI     3$              ;N:C = 0101 ?
6133  020126  001002                          BNE     3$
6134  020130  102401                          BVS     3$
6135  020132  103401                          BCS     4$
6136
6137  020134  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
6138
6139  020136  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6140  020140  001401                          BEQ     TST274          ;;BR IF YES
6141
6142  020142  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
6143
6144                                  ;;**********************************************************************
6145                                  ;*TEST 274       INC DMO TEST - N:C = 0100
6146                                  ;;**********************************************************************
6147  020144                          TST274:
6148  020144  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6149  020146  012700  000274                  MOV     #274,R0         ;;LOAD R0 WITH TEST NUMBER
6150  020152  013701  020172                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6151  020156  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6152  020162  012703  077777                  MOV     #77777,R3       ;[DEST] = 77777
6153  020166  000257                          CCC                     ;CLEAR CODES
6154  020170  000264                          264                     ;N:C = 0100
6155
6156  020172  005203                  2$:     INC     R3              ;TEST THE INC
6157
6158  020174  100003                          BPL     3$              ;N:C = 1010 ?
6159  020176  001402                          BEQ     3$
6160  020200  102001                          BVC     3$
6161  020202  103001                          BCC     4$
6162
6163  020204  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
6164
6165  020206  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
6166  020210  001401                          BEQ     TST275          ;;BR IF YES
6167
6168  020212  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
6169
6170                                  ;;**********************************************************************
6171                                  ;*TEST 275       DEC DMO TEST - N:C = 1011
6172                                  ;;**********************************************************************
6173  020214                          TST275:
6174  020214  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6175  020216  012700  000275                  MOV     #275,R0         ;;LOAD R0 WITH TEST NUMBER
6176  020222  013701  020240                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6177  020226  005004                          CLR     R4              ;RESULT S / B = 000000
6178  020230  012703  000001                  MOV     #1,R3           ;[DEST] = 1
6179  020234  000257                          CCC                     ;CLEAR CODES
6180  020236  000273                          273                     ;N:C = 1011
6181
6182  020240  005303                  2$:     DEC     R3              ;TEST THE DEC
6183
6184  020242  100403                          BMI     3$              ;N:C = 0101 ?
```

```
 6185  020244  001002                    BNE     3$
 6186  020246  102401                    BVS     3$
 6187  020250  103401                    BCS     4$
 6188
 6189  020252  104002            3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6190
 6191  020254  020403            4$:     CMP     R4,R3           ;RESULT OK ?
 6192  020256  001401                    BEQ     TST276          ;;BR IF YES
 6193
 6194  020260  104002            5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6195
 6196                            ;;**********************************************************************
 6197                            ;*TEST 276       DEC DM0 TEST - N:C = 1100
 6198                            ;;**********************************************************************
 6199  020262                    TST276:
 6200  020262  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6201  020264  012700  000276            MOV     #276,R0         ;;LOAD R0 WITH TEST NUMBER
 6202  020270  013701  020310            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6203  020274  012704  077777            MOV     #77777,R4       ;RESULT S / B = 77777
 6204  020300  012703  100000            MOV     #100000,R3      ;[DEST] = 100000
 6205  020304  000257                    CCC                     ;CLEAR CODES
 6206  020306  000274                    274                     ;N:C = 1100
 6207
 6208  020310  005303            2$:     DEC     R3              ;TEST THE DEC
 6209
 6210  020312  100403                    BMI     3$              ;N:C = 0010 ?
 6211  020314  001402                    BEQ     3$
 6212  020316  102001                    BVC     3$
 6213  020320  103001                    BCC     4$
 6214
 6215  020322  104002            3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6216
 6217  020324  020403            4$:     CMP     R4,R3           ;RESULT OK ?
 6218  020326  001401                    BEQ     TST277          ;;BR IF YES
 6219
 6220  020330  104002            5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6221
 6222                            ;;**********************************************************************
 6223                            ;*TEST 277       DEC DM0 TEST - N:C = 0000
 6224                            ;;**********************************************************************
 6225  020332                    TST277:
 6226  020332  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6227  020334  012700  000277            MOV     #277,R0         ;;LOAD R0 WITH TEST NUMBER
 6228  020340  013701  020354            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6229  020344  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
 6230  020350  005003                    CLR     R3              ;[DEST] = 000000
 6231  020352  000257                    CCC                     ;CLEAR CODES
 6232
 6233  020354  005303            2$:     DEC     R3              ;TEST THE DEC
```

```
6234
6235  020356  100003                  BPL     3$              ;N:C = 1000 ?
6236  020360  001402                  BEQ     3$
6237  020362  102401                  BVS     3$
6238  020364  103001                  BCC     4$
6239
6240  020366  104002          3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
6241
6242  020370  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6243  020372  001401                  BEQ     TST300          ;;BR IF YES
6244
6245  020374  104002          5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
6246
6247                          ;;**************************************************************
6248                          ;*TEST 300       ASL DMO TEST - N:C = 1000
6249                          ;;**************************************************************
6250  020376                  TST300:
6251  020376  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6252  020400  012700  000300          MOV     #300,R0         ;;LOAD R0 WITH TEST NUMBER
6253  020404  013701  020422          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6254  020410  005004                  CLR     R4              ;RESULT S / B = 000000
6255  020412  012703  100000          MOV     #100000,R3      ;[DEST] = 100000
6256  020416  000257                  CCC                     ;CLEAR CODES
6257  020420  000270                  SEN                     ;N:C = 1000
6258
6259  020422  006303          2$:     ASL     R3              ;TEST THE ASL
6260
6261  020424  100403                  BMI     3$              ;N:C = 0111 ?
6262  020426  001002                  BNE     3$
6263  020430  102001                  BVC     3$
6264  020432  103401                  BCS     4$
6265
6266  020434  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6267
6268  020436  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6269  020440  001401                  BEQ     TST301          ;;BR IF YES
6270
6271  020442  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6272
6273                          ;;**************************************************************
6274                          ;*TEST 301       ASL DMO TEST - N:C = 0101
6275                          ;;**************************************************************
6276  020444                  TST301:
6277  020444  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6278  020446  012700  000301          MOV     #301,R0         ;;LOAD R0 WITH TEST NUMBER
6279  020452  013701  020472          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6280  020456  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
6281  020462  012703  040000          MOV     #40000,R3       ;[DEST] = 40000
6282  020466  000257                  CCC                     ;CLEAR CODES
6283  020470  000265                  265                     ;N:C = 0101
6284
6285  020472  006303          2$:     ASL     R3              ;TEST THE ASL
6286
6287  020474  100003                  BPL     3$              ;N:C = 1010 ?
6288  020476  001402                  BEQ     3$
6289  020500  102001                  BVC     3$
```

```
6290  020502  103001                  BCC     4$
6291
6292  020504  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6293
6294  020506  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6295  020510  001401                  BEQ     TST302          ;;BR IF YES
6296
6297  020512  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6298
6299                          ;;*******************************************************************
6300                          ;*TEST 302       ASL DM0 TEST - N:C = 0010
6301                          ;;*******************************************************************
6302  020514                  TST302:
6303  020514  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6304  020516  012700  000302          MOV     #302,R0         ;;LOAD R0 WITH TEST NUMBER
6305  020522  013701  020536          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6306  020526  005004                  CLR     R4              ;RESULT S  /  B = 000000
6307  020530  005003                  CLR     R3              ;[DEST] = 000000
6308  020532  000257                  CCC                     ;CLEAR CODES
6309  020534  000262                  SEV                     ;N:C = 0010
6310
6311  020536  006303          2$:     ASL     R3              ;TEST THE ASL
6312
6313  020540  100403                  BMI     3$              ;N:C = 0100 ?
6314  020542  001002                  BNE     3$
6315  020544  102401                  BVS     3$
6316  020546  103001                  BCC     4$
6317
6318  020550  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
6319
6320  020552  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6321  020554  001401                  BEQ     TST303          ;;BR IF YES
6322
6323  020556  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
6324
6325                          ;;*******************************************************************
6326                          ;*TEST 303       ROL DM0 TEST - N:C = 1101
6327                          ;;*******************************************************************
6328  020560                  TST303:
6329  020560  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6330  020562  012700  000303          MOV     #303,R0         ;;LOAD R0 WITH TEST NUMBER
6331  020566  013701  020606          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6332  020572  012704  052525          MOV     #52525,R4       ;RESULT S  /  B = 52525
6333  020576  012703  125252          MOV     #125252,R3      ;[DEST] = 125252
6334  020602  000257                  CCC                     ;CLEAR CODES
6335  020604  000275                  275                     ;N:C = 1101
6336
6337  020606  006103          2$:     ROL     R3              ;TEST THE ROL
6338
6339  020610  100403                  BMI     3$              ;N:C = 0011 ?
6340  020612  001402                  BEQ     3$
6341  020614  102001                  BVC     3$
6342  020616  103401                  BCS     4$
6343
6344  020620  104002          3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
6345
```

```
                                4$:     CMP     R4,R3           ;RESULT OK ?
                                        BEQ     TST304          ;;BR IF YES

                                5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT

                                ;;*******************************************************************
                                ;*TEST 304       ROL DM0 TEST - N:C = 0101
                                ;;*******************************************************************
                                TST304:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #304,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #125253,R4      ;RESULT S / B = 125253
                                        MOV     #52525,R3       ;[DEST] = 52525
                                        CCC                     ;CLEAR CODES
                                        265                     ;N:C = 0101

                                2$:     ROL     R3              ;TEST THE ROL

                                        BPL     3$              ;N:C = 1010 ?
                                        BEQ     3$
                                        BVC     3$
                                        BCC     4$

                                3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
                                4$:     CMP     R4,R3           ;RESULT OK ?
                                        BEQ     TST305          ;;BR IF YES

                                5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT

                                ;;*******************************************************************
                                ;*TEST 305       ROL DM0 TEST - N:C = 0010
                                ;;*******************************************************************
                                TST305:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #305,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        CLR     R4              ;RESULT S / B = 000000
                                        CLR     R3              ;[DEST] = 000000
                                        CCC                     ;CLEAR CODES
                                        SEV                     ;N:C = 0010

                                2$:     ROL     R3              ;TEST THE ROL

                                        BMI     3$              ;N:C = 0100 ?
                                        BNE     3$
                                        BVS     3$
                                        BCC     4$

                                3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY

                                4$:     CMP     R4,R3           ;RESULT OK ?
                                        BEQ     TST306          ;;BR IF YES

                                5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
```

```
                                        ;;*******************************************************************
                                        ;*TEST 306      ADC DMO TEST - N:C = 0101
                                        ;;*******************************************************************
020744                                  TST306:
020744  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020746  012700  000306                          MOV     #306,R0         ;;LOAD R0 WITH TEST NUMBER
020752  013701  020772                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020756  012704  100000                          MOV     #100000,R4      ;RESULT S / B = 100000
020762  012703  077777                          MOV     #77777,R3       ;[DEST] = 77777
020766  000257                                  CCC                     ;CLEAR CODES
020770  000265                                  265                     ;N:C = 0101

020772  005503                          2$:     ADC     R3              ;TEST THE ADC

020774  100003                                  BPL     3$              ;N:C = 1010 ?
020776  001402                                  BEQ     3$
021000  102001                                  BVC     3$
021002  103001                                  BCC     4$

021004  104002                          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY

021006  020403                          4$:     CMP     R4,R3           ;RESULT OK ?
021010  001401                                  BEQ     TST307          ;;BR IF YES

021012  104002                          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

                                        ;;*******************************************************************
                                        ;*TEST 307      ADC DMO TEST - N:C = 1011
                                        ;;*******************************************************************
021014                                  TST307:
021014  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
021016  012700  000307                          MOV     #307,R0         ;;LOAD R0 WITH TEST NUMBER
021022  013701  021040                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
021026  005004                                  CLR     R4              ;RESULT S / B = 000000
021030  012703  177777                          MOV     #-1,R3          ;[DEST] = 177777
021034  000257                                  CCC                     ;CLEAR CODES
021036  000273                                  273                     ;N:C = 1011

021040  005503                          2$:     ADC     R3              ;TEST THE ADC

021042  100403                                  BMI     3$              ;N:C = 0101 ?
021044  001002                                  BNE     3$
021046  102401                                  BVS     3$
021050  103401                                  BCS     4$

021052  104002                          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY

021054  020403                          4$:     CMP     R4,R3           ;RESULT OK ?
021056  001401                                  BEQ     TST310          ;;BR IF YES

021060  104002                          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

                                        ;;*******************************************************************
                                        ;*TEST 310      ADC DMO TEST - N:C = 1010
                                        ;;*******************************************************************
021062                                  TST310:
```

```
                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                        MOV     #310,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #-1,R4          ;RESULT S / B = 177777
                        MOV     #-1,R3          ;[DEST] = 177777
                        CCC                     ;CLEAR CODES
                        272                     ;N:C = 1010

2$:                     ADC     R3              ;TEST THE ADC

                        BPL     3$              ;N:C = 1000 ?
                        BEQ     3$
                        BVS     3$
                        BCC     4$

3$:                     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY

4$:                     CMP     R4,R3           ;RESULT OK ?
                        BEQ     TST311          ;;BR IF YES

5$:                     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

;;**********************************************************************
;*TEST 311       SBC DM0 TEST - N:C = 1011
;;**********************************************************************
TST311:
                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                        MOV     #311,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        CLR     R4              ;RESULT S / B = 000000
                        MOV     #1,R3           ;[DEST] = +1
                        CCC                     ;CLEAR CODES
                        273                     ;N:C = 1011

2$:                     SBC     R3              ;TEST THE SBC

                        BMI     3$              ;N:C = 0100 ?
                        BNE     3$
                        BVS     3$
                        BCC     4$

3$:                     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY

4$:                     CMP     R4,R3           ;RESULT OK ?
                        BEQ     TST312          ;;BR IF YES

5$:                     ERROR   2               ;SBC DELIVERED THE WRONG RESULT

;;**********************************************************************
;*TEST 312       SBC DM0 TEST - N:C = 0101
;;**********************************************************************
TST312:
                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                        MOV     #312,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #077777,R4      ;RESULT S / B = 077777
```

```
6514  021216  012703  100000          MOV     #100000,R3      ;[DEST] = 100000
6515  021222  000257                  CCC                     ;CLEAR CODES
6516  021224  000265                  265                     ;N:C = 0101
6517
6518  021226  005603          2$:     SBC     R3              ;TEST THE SBC
6519
6520  021230  100403                  BMI     3$              ;N:C = 0010 ?
6521  021232  001402                  BEQ     3$
6522  021234  102001                  BVC     3$
6523  021236  103001                  BCC     4$
6524
6525  021240  104002          3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6526
6527  021242  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6528  021244  001401                  BEQ     TST313          ;;BR IF YES
6529
6530  021246  104002          5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6531
6532                          ;;**********************************************************************
6533                          ;*TEST 313       SBC DM0 TEST - N:C = 1110
6534                          ;;**********************************************************************
6535  021250                  TST313:
6536  021250  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6537  021252  012700  000313          MOV     #313,R0         ;;LOAD R0 WITH TEST NUMBER
6538  021256  013701  021276          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6539  021262  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
6540  021266  012703  000001          MOV     #1,R3           ;[DEST] = 1
6541  021272  000257                  CCC                     ;CLEAR CODES
6542  021274  000276                  276                     ;N:C = 1110
6543
6544  021276  005603          2$:     SBC     R3              ;TEST THE SBC
6545
6546  021300  100403                  BMI     3$              ;N:C = 0000 ?
6547  021302  001402                  BEQ     3$
6548  021304  102401                  BVS     3$
6549  021306  103001                  BCC     4$
6550
6551  021310  104002          3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
6552
6553  021312  020403          4$:     CMP     R4,R3           ;RESULT OK ?
6554  021314  001401                  BEQ     TST314          ;;BR IF YES
6555
6556  021316  104002          5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
6557
6558                          ;;**********************************************************************
6559                          ;*TEST 314       SBC DM0 TEST - N:C = 0111
6560                          ;;**********************************************************************
6561  021320                  TST314:
6562  021320  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6563  021322  012700  000314          MOV     #314,R0         ;;LOAD R0 WITH TEST NUMBER
6564  021326  013701  021344          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6565  021332  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
6566  021336  005003                  CLR     R3              ;[DEST] = 000000
6567  021340  000257                  CCC                     ;CLEAR CODES
6568  021342  000267                  267                     ;N:C = 0111
6569
```

```
 6570  021344  005603                 2$:     SBC     R3              ;TEST THE SBC
 6571
 6572  021346  100003                         BPL     3$              ;N:C = 1001 ?
 6573  021350  001402                         BEQ     3$
 6574  021352  102401                         BVS     3$
 6575  021354  103401                         BCS     4$
 6576
 6577  021356  104002                 3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6578
 6579  021360  020403                 4$:     CMP     R4,R3           ;RESULT OK ?
 6580  021362  001401                         BEQ     TST315          ;;BR IF YES
 6581
 6582  021364  104002                 5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6583
 6584                                 ;;*********************************************************************
 6585                                 ;*TEST 315       TST DM1 TEST - N:C = 1011
 6586                                 ;;*********************************************************************
 6587  021366                         TST315:
 6588  021366  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6589  021370  012700  000315                 MOV     #315,R0         ;;LOAD R0 WITH TEST NUMBER
 6590  021374  013701  021414                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6591  021400  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6592  021404  005004                         CLR     R4              ;RESULT S / B = 000000
 6593  021406  005012                         CLR     (R2)            ;[DEST] = 000000
 6594  021410  000257                         CCC                     ;CLEAR CODES
 6595  021412  000273                         273                     ;N:C=1011
 6596
 6597  021414  005712                 2$:     TST     (R2)            ;TEST THE TST
 6598
 6599  021416  100403                         BMI     3$              ;N:C = 0100 ?
 6600  021420  001002                         BNE     3$
 6601  021422  102401                         BVS     3$
 6602  021424  103001                         BCC     4$
 6603
 6604  021426  104001                 3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
 6605
 6606  021430  020412                 4$:     CMP     R4,(R2)         ;RESULT OK ?
 6607  021432  001402                         BEQ     TST316          ;;BR IF YES
 6608
 6609  021434  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 6610  021436  104001                 5$:     ERROR   1               ;TST ALTERED THE [DEST]
 6611
 6612                                 ;;*********************************************************************
 6613                                 ;*TEST 316       TST DM1 TEST - N:C = 0100
 6614                                 ;;*********************************************************************
 6615  021440                         TST316:
 6616  021440  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6617  021442  012700  000316                 MOV     #316,R0         ;;LOAD R0 WITH TEST NUMBER
 6618  021446  013701  021472                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6619  021452  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6620  021456  005004                         CLR     R4
 6621  021460  005104                         COM     R4              ;RESULT S / B = 177777
 6622  021462  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
 6623  021466  000257                         CCC                     ;CLEAR CODES
 6624  021470  000264                         264                     ;N:C=0100
 6625
```

```
6626  021472  005712                  2$:     TST     (R2)            ;TEST THE TST
6627
6628  021474  100003                          BPL     3$              ;N:C = 1000 ?
6629  021476  001402                          BEQ     3$
6630  021500  102401                          BVS     3$
6631  021502  103001                          BCC     4$
6632
6633  021504  104001                  3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
6634
6635  021506  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6636  021510  001402                          BEQ     TST317          ;;BR IF YES
6637
6638  021512  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6639  021514  104001                  5$:     ERROR   1               ;TST ALTERED THE [DEST]
6640
6641                                  ;;**********************************************************************
6642                                  ;*TEST 317       CLR DM1 TEST - N:C = 1011
6643                                  ;;**********************************************************************
6644  021516                          TST317:
6645  021516  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6646  021520  012700  000317                  MOV     #317,R0         ;;LOAD R0 WITH TEST NUMBER
6647  021524  013701  021560                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6648                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT6
6649  021530  032737  000100  063234          BIT     #BIT6,@#BPTLOC  ;BREAKPOINT HALT SET ??
6650  021536  001401                          BEQ     .+4             ;BR IF NOT
6651  021540  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
6652  021542  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6653  021546  005004                          CLR     R4              ;RESULT S / B = 000000
6654  021550  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
6655  021554  000257                          CCC                     ;CLEAR CODES
6656  021556  000273                          273                     ;N:C = 1011
6657
6658  021560  005012                  2$:     CLR     (R2)            ;TEST THE CLR
6659
6660  021562  100403                          BMI     3$              ;N:C = 0100 ?
6661  021564  001002                          BNE     3$
6662  021566  102401                          BVS     3$
6663  021570  103001                          BCC     4$
6664
6665  021572  104001                  3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
6666
6667  021574  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6668  021576  001402                          BEQ     TST320          ;;BR IF YES
6669
6670  021600  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6671  021602  104001                  5$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
6672
6673                                  ;;**********************************************************************
6674                                  ;*TEST 320       CLR DM2 TEST - N:C = 0000
6675                                  ;;**********************************************************************
6676  021604                          TST320:
6677  021604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6678  021606  012700  000320                  MOV     #320,R0         ;;LOAD R0 WITH TEST NUMBER
6679  021612  013701  021632                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6680  021616  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6681  021622  005004                          CLR     R4              ;RESULT S / B = 000000
```

```
6682  021624  013712  063324          MOV     @#DWTA+2,(R2)   ;[DEST] = 177777
6683  021630  000257                  CCC                     ;CLEAR CODES
6684
6685  021632  005022          2$:     CLR     (R2)+           ;TEST THE CLR
6686
6687  021634  100403                  BMI     3$              ;N:C = 0100 ?
6688  021636  001002                  BNE     3$
6689  021640  102401                  BVS     3$
6690  021642  103001                  BCC     4$
6691
6692  021644  104001          3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
6693
6694  021646  022702  063314  4$:     CMP     #MBUF0+2,R2     ;DID CLR INCREMENT DEST REG
6695  021652  001401                  BEQ     6$              ;BR IF YES
6696
6697  021654  104005          5$:     ERROR   5               ;CLR FAILED TO UPDATE DEST REG
6698
6699  021656  020442          6$:     CMP     R4,-(R2)                ;RESULT OK ?
6700  021660  001402                  BEQ     TST321          ;;BR IF YES
6701
6702  021662  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6703  021664  104001          7$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
6704
6705                          ;;**********************************************************************
6706                          ;*TEST 321       COM DM1 TEST - N:C = 0110
6707                          ;;**********************************************************************
6708  021666                  TST321:
6709  021666  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6710  021670  012700  000321          MOV     #321,R0         ;;LOAD R0 WITH TEST NUMBER
6711  021674  013701  021720          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6712  021700  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6713  021704  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6714  021710  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
6715  021714  000257                  CCC                     ;CLEAR CODES
6716  021716  000266                  266                     ;N:C = 0110
6717
6718  021720  005112          2$:     COM     (R2)            ;TEST THE CLR
6719
6720  021722  100003                  BPL     3$              ;N:C = 1001 ?
6721  021724  001402                  BEQ     3$
6722  021726  102401                  BVS     3$
6723  021730  103401                  BCS     4$
6724
6725  021732  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6726  021734  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6727  021736  001402                  BEQ     TST322          ;;BR IF YES
6728
6729  021740  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6730  021742  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6731
6732                          ;;**********************************************************************
6733                          ;*TEST 322       COM DM1 TEST - N:C = 1001
6734                          ;;**********************************************************************
6735  021744                  TST322:
6736  021744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6737  021746  012700  000322          MOV     #322,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
6738  021752  013701  021774          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6739  021756  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6740  021762  005004                  CLR     R4              ;RESULT S / B = 000000
6741  021764  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
6742  021770  000257                  CCC                     ;CLEAR CODES
6743  021772  000271                  271                     ;N:C = 1001
6744
6745  021774  005112          2$:     COM     (R2)            ;TEST THE COM
6746
6747  021776  100403                  BMI     3$              ;N:C = 0101 ?
6748  022000  001002                  BNE     3$
6749  022002  102401                  BVS     3$
6750  022004  103401                  BCS     4$
6751
6752  022006  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6753  022010  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6754  022012  001402                  BEQ     TST323          ;;BR IF YES
6755
6756  022014  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6757  022016  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6758
6759                          ;;**************************************************************
6760                          ;*TEST 323       INC DM1 TEST - N:C = 1011
6761                          ;;**************************************************************
6762  022020                  TST323:
6763  022020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6764  022022  012700  000323          MOV     #323,R0         ;;LOAD R0 WITH TEST NUMBER
6765  022026  013701  022050          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6766  022032  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6767  022036  005004                  CLR     R4              ;RESULT S / B = 000000
6768  022040  012712  177777          MOV     #-1,(R2)                ;[DEST] = 177777
6769  022044  000257                  CCC                     ;CLEAR CODES
6770  022046  000273                  273                     ;N:C = 1011
6771
6772  022050  005212          2$:     INC     (R2)            ;TEST THE INC
6773
6774  022052  100403                  BMI     3$              ;N:C = 0101 ?
6775  022054  001002                  BNE     3$
6776  022056  102401                  BVS     3$
6777  022060  103401                  BCS     4$
6778
6779  022062  104001          3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6780  022064  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6781  022066  001402                  BEQ     TST324          ;;BR IF YES
6782
6783  022070  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6784  022072  104001          5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6785
6786                          ;;**************************************************************
6787                          ;*TEST 324       INC DM1 TEST - N:C = 0100
6788                          ;;**************************************************************
6789  022074                  TST324:
6790  022074  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6791  022076  012700  000324          MOV     #324,R0         ;;LOAD R0 WITH TEST NUMBER
6792  022102  013701  022126          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6793  022106  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
6794  022112  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
6795  022116  012712  077777          MOV     #77777,(R2)     ;[DEST] = 77777
6796  022122  000257                  CCC                     ;CLEAR CODES
6797  022124  000264                  264                     ;N:C = 0100
6798
6799  022126  005212          2$:     INC     (R2)            ;TEST THE INC
6800
6801  022130  100003                  BPL     3$              ;N:C = 1010 ?
6802  022132  001402                  BEQ     3$
6803  022134  102001                  BVC     3$
6804  022136  103001                  BCC     4$
6805
6806  022140  104001          3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6807  022142  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6808  022144  001402                  BEQ     TST325          ;;BR IF YES
6809
6810  022146  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6811  022150  104001          5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6812
6813                          ;;**********************************************************************
6814                          ;*TEST 325       DEC DM1 TEST - N:C = 1011
6815                          ;;**********************************************************************
6816  022152                  TST325:
6817  022152  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6818  022154  012700  000325          MOV     #325,R0         ;;LOAD R0 WITH TEST NUMBER
6819  022160  013701  022202          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6820  022164  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6821  022170  005004                  CLR     R4              ;RESULT S / B = 000000
6822  022172  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
6823  022176  000257                  CCC                     ;CLEAR CODES
6824  022200  000273                  273                     ;N:C = 1011
6825
6826  022202  005312          2$:     DEC     (R2)            ;TEST THE DEC
6827
6828  022204  100403                  BMI     3$              ;N:C = 0101 ?
6829  022206  001002                  BNE     3$
6830  022210  102401                  BVS     3$
6831  022212  103401                  BCS     4$
6832
6833  022214  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6834  022216  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6835  022220  001402                  BEQ     TST326          ;;BR IF YES
6836
6837  022222  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6838  022224  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6839
6840                          ;;**********************************************************************
6841                          ;*TEST 326       DEC DM1 TEST - N:C = 1100
6842                          ;;**********************************************************************
6843  022226                  TST326:
6844  022226  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6845  022230  012700  000326          MOV     #326,R0         ;;LOAD R0 WITH TEST NUMBER
6846  022234  013701  022260          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6847  022240  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6848  022244  012704  077777          MOV     #77777,R4       ;RESULT S / B = 77777
6849  022250  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
```

```
 6850  022254  000257                    CCC                     ;CLEAR CODES
 6851  022256  000274                    274                     ;N:C = 1100
 6852
 6853  022260  005312            2$:     DEC     (R2)            ;TEST THE DEC
 6854
 6855  022262  100403                    BMI     3$              ;N:C = 0010 ?
 6856  022264  001402                    BEQ     3$
 6857  022266  102001                    BVC     3$
 6858  022270  103001                    BCC     4$
 6859
 6860  022272  104001            3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6861  022274  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
 6862  022276  001402                    BEQ     TST327          ;;BR IF YES
 6863
 6864  022300  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 6865  022302  104001            5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6866
 6867                            ;;**********************************************************************
 6868                            ;*TEST 327      DEC DM1 TEST - N:C = 0000
 6869                            ;;**********************************************************************
 6870  022304                    TST327:
 6871  022304  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6872  022306  012700  000327            MOV     #327,R0         ;;LOAD R0 WITH TEST NUMBER
 6873  022312  013701  022332            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6874  022316  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6875  022322  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
 6876  022326  005012                    CLR     (R2)            ;[DEST] = 000000
 6877  022330  000257                    CCC                     ;CLEAR CODES
 6878
 6879  022332  005312            2$:     DEC     (R2)            ;TEST THE DEC
 6880
 6881  022334  100003                    BPL     3$              ;N:C = 1000 ?
 6882  022336  001402                    BEQ     3$
 6883  022340  102401                    BVS     3$
 6884  022342  103001                    BCC     4$
 6885
 6886  022344  104001            3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6887  022346  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
 6888  022350  001402                    BEQ     TST330          ;;BR IF YES
 6889
 6890  022352  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 6891  022354  104001            5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6892
 6893                            ;;**********************************************************************
 6894                            ;*TEST 330      ASL DM1 TEST - N:C = 1000
 6895                            ;;**********************************************************************
 6896  022356                    TST330:
 6897  022356  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6898  022360  012700  000330            MOV     #330,R0         ;;LOAD R0 WITH TEST NUMBER
 6899  022364  013701  022406            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6900  022370  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6901  022374  005004                    CLR     R4              ;RESULT S / B = 000000
 6902  022376  012712  100000            MOV     #100000,(R2)    ;[DEST] = 100000
 6903  022402  000257                    CCC                     ;CLEAR CODES
 6904  022404  000270                    SEN                     ;N:C = 1000
 6905
```

```
 6906  022406  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6907
 6908  022410  100403                        BMI     3$              ;N:C = 0111 ?
 6909  022412  001002                        BNE     3$
 6910  022414  102001                        BVC     3$
 6911  022416  103401                        BCS     4$
 6912
 6913  022420  104001                3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6914  022422  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6915  022424  001402                        BEQ     TST331          ;;BR IF YES
 6916
 6917  022426  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6918  022430  104001                5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6919
 6920                                ;;**********************************************************************
 6921                                ;*TEST 331       ASL DM1 TEST - N:C = 0101
 6922                                ;;**********************************************************************
 6923  022432                        TST331:
 6924  022432  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6925  022434  012700  000331                MOV     #331,R0         ;;LOAD R0 WITH TEST NUMBER
 6926  022440  013701  022464                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6927  022444  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6928  022450  012704  100000                MOV     #100000,R4      ;RESULT S / B = 100000
 6929  022454  012712  040000                MOV     #40000,(R2)     ;[DEST] = 40000
 6930  022460  000257                        CCC                     ;CLEAR CODES
 6931  022462  000265                        265                     ;N:C = 0101
 6932
 6933  022464  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6934
 6935  022466  100003                        BPL     3$              ;N:C = 1010 ?
 6936  022470  001402                        BEQ     3$
 6937  022472  102001                        BVC     3$
 6938  022474  103001                        BCC     4$
 6939
 6940  022476  104001                3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6941  022500  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6942  022502  001402                        BEQ     TST332          ;;BR IF YES
 6943
 6944  022504  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6945  022506  104001                5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6946
 6947                                ;;**********************************************************************
 6948                                ;*TEST 332       ASL DM1 TEST - N:C = 0010
 6949                                ;;**********************************************************************
 6950  022510                        TST332:
 6951  022510  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6952  022512  012700  000332                MOV     #332,R0         ;;LOAD R0 WITH TEST NUMBER
 6953  022516  013701  022536                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6954  022522  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6955  022526  005004                        CLR     R4              ;RESULT S / B = 000000
 6956  022530  005012                        CLR     (R2)            ;[DEST] = 000000
 6957  022532  000257                        CCC                     ;CLEAR CODES
 6958  022534  000262                        SEV                     ;N:C = 0010
 6959
 6960  022536  006312                2$:     ASL     (R2)            ;TEST THE ASL
 6961
```

```
                        BMI     3$              ;N:C = 0100 ?
                        BNE     3$
                        BVS     3$
                        BCC     4$

                3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
                4$:     CMP     R4,(R2)         ;RESULT OK ?
                        BEQ     TST333          ;;BR IF YES

                        MOV     (R2),R3         ;GET THE WAS DATA
                5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT

                ;;*****************************************************************
                ;*TEST 333      ROL DM1 TEST - N:C = 1101
                ;;*****************************************************************
                TST333:
                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                        MOV     #333,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                        MOV     #52525,R4       ;RESULT S / B = 52525
                        MOV     #125252,(R2)    ;[DEST] = 125252
                        CCC                     ;CLEAR CODES
                        275                     ;N:C = 1101

                2$:     ROL     (R2)            ;TEST THE ROL

                        BMI     3$              ;N:C = 0011 ?
                        BEQ     3$
                        BVC     3$
                        BCS     4$

                3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
                4$:     CMP     R4,(R2)         ;RESULT OK ?
                        BEQ     TST334          ;;BR IF YES

                        MOV     (R2),R3         ;GET THE WAS DATA
                5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT

                ;;*****************************************************************
                ;*TEST 334      ROL DM1 TEST - N:C = 0101
                ;;*****************************************************************
                TST334:
                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                        MOV     #334,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                        MOV     #125253,R4      ;RESULT S / B = 125253
                        MOV     #52525,(R2)     ;[DEST] = 52525
                        CCC                     ;CLEAR CODES
                        265                     ;N:C = 0101

                2$:     ROL     (R2)            ;TEST THE ROL

                        BPL     3$              ;N:C = 1010 ?
                        BEQ     3$
```

```
7018  022700  102001                BVC     3$
7019  022702  103001                BCC     4$
7020
7021  022704  104001          3$:   ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7022  022706  020412          4$:   CMP     R4,(R2)         ;RESULT OK ?
7023  022710  001402                BEQ     TST335          ;;BR IF YES
7024
7025  022712  011203                MOV     (R2),R3         ;GET THE WAS DATA
7026  022714  104001          5$:   ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7027
7028                          ;;**********************************************************************
7029                          ;*TEST 335       ROL DM1 TEST - N:C = 0010
7030                          ;;**********************************************************************
7031  022716                  TST335:
7032  022716  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7033  022720  012700  000335        MOV     #335,R0         ;;LOAD R0 WITH TEST NUMBER
7034  022724  013701  022744        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7035  022730  012702  063312        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7036  022734  005004                CLR     R4              ;RESULT S / B = 000000
7037  022736  005012                CLR     (R2)            ;[DEST] = 000000
7038  022740  000257                CCC                     ;CLEAR CODES
7039  022742  000262                SEV                     ;N:C = 0010
7040
7041  022744  006112          2$:   ROL     (R2)            ;TEST THE ROL
7042
7043  022746  100403                BMI     3$              ;N:C = 0100 ?
7044  022750  001002                BNE     3$
7045  022752  102401                BVS     3$
7046  022754  103001                BCC     4$
7047
7048  022756  104001          3$:   ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
7049  022760  020412          4$:   CMP     R4,(R2)         ;RESULT OK ?
7050  022762  001402                BEQ     TST336          ;;BR IF YES
7051
7052  022764  011203                MOV     (R2),R3         ;GET THE WAS DATA
7053  022766  104001          5$:   ERROR   1               ;ROL DELIVERED THE WRONG RESULT
7054
7055                          ;;**********************************************************************
7056                          ;*TEST 336       ADC DM1 TEST - N:C = 0101
7057                          ;;**********************************************************************
7058  022770                  TST336:
7059  022770  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7060  022772  012700  000336        MOV     #336,R0         ;;LOAD R0 WITH TEST NUMBER
7061  022776  013701  023022        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7062  023002  012702  063312        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7063  023006  012704  100000        MOV     #100000,R4      ;RESULT S / B = 100000
7064  023012  012712  077777        MOV     #77777,(R2)     ;[DEST] = 77777
7065  023016  000257                CCC                     ;CLEAR CODES
7066  023020  000265                265                     ;N:C = 0101
7067
7068  023022  005512          2$:   ADC     (R2)            ;TEST THE ADC
7069
7070  023024  100003                BPL     3$              ;N:C = 1010 ?
7071  023026  001402                BEQ     3$
7072  023030  102001                BVC     3$
7073  023032  103001                BCC     4$
```

```
7074
7075  023034  104001                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
7076  023036  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
7077  023040  001402                        BEQ     TST337          ;;BR IF YES
7078
7079  023042  011203                        MOV     (R2),R3         ;GET THE WAS DATA
7080  023044  104001                5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
7081
7082                                ;;**********************************************************************
7083                                ;*TEST 337       ADC DM1 TEST - N:C = 1011
7084                                ;;**********************************************************************
7085  023046                        TST337:
7086  023046  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7087  023050  012700  000337                MOV     #337,R0         ;;LOAD R0 WITH TEST NUMBER
7088  023054  013701  023076                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7089  023060  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7090  023064  005004                        CLR     R4              ;RESULT S  /  B = 000000
7091  023066  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
7092  023072  000257                        CCC                     ;CLEAR CODES
7093  023074  000273                        273                     ;N:C = 1011
7094
7095  023076  005512                2$:     ADC     (R2)            ;TEST THE ADC
7096
7097  023100  100403                        BMI     3$              ;N:C = 0101 ?
7098  023102  001002                        BNE     3$
7099  023104  102401                        BVS     3$
7100  023106  103401                        BCS     4$
7101
7102  023110  104001                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
7103  023112  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
7104  023114  001402                        BEQ     TST340          ;;BR IF YES
7105
7106  023116  011203                        MOV     (R2),R3         ;GET THE WAS DATA
7107  023120  104001                5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
7108
7109                                ;;**********************************************************************
7110                                ;*TEST 340       ADC DM1 TEST - N:C = 1010
7111                                ;;**********************************************************************
7112  023122                        TST340:
7113  023122  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7114  023124  012700  000340                MOV     #340,R0         ;;LOAD R0 WITH TEST NUMBER
7115  023130  013701  023154                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7116  023134  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7117  023140  012704  177777                MOV     #-1,R4          ;RESULT S  /  B = 177777
7118  023144  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
7119  023150  000257                        CCC                     ;CLEAR CODES
7120  023152  000272                        272                     ;N:C = 1010
7121
7122  023154  005512                2$:     ADC     (R2)            ;TEST THE ADC
7123
7124  023156  100003                        BPL     3$              ;N:C = 1000 ?
7125  023160  001402                        BEQ     3$
7126  023162  102401                        BVS     3$
7127  023164  103001                        BCC     4$
7128
7129  023166  104001                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
```

```
7130  023170  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7131  023172  001402                          BEQ     TST341          ;;BR IF YES
7132
7133  023174  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7134  023176  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
7135
7136                                  ;;**********************************************************************
7137                                  ;*TEST 341       SBC DM1 TEST - N:C = 1011
7138                                  ;;**********************************************************************
7139  023200                          TST341:
7140  023200  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7141  023202  012700  000341                  MOV     #341,R0         ;;LOAD R0 WITH TEST NUMBER
7142  023206  013701  023230                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7143  023212  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7144  023216  005004                          CLR     R4              ;RESULT S / B = 000000
7145  023220  012712  000001                  MOV     #1,(R2)         ;[DEST] = +1
7146  023224  000257                          CCC                     ;CLEAR CODES
7147  023226  000273                          273                     ;N:C = 1011
7148
7149  023230  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7150
7151  023232  100403                          BMI     3$              ;N:C = 0100 ?
7152  023234  001002                          BNE     3$
7153  023236  102401                          BVS     3$
7154  023240  103001                          BCC     4$
7155
7156  023242  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7157  023244  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7158  023246  001402                          BEQ     TST342          ;;BR IF YES
7159
7160  023250  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7161  023252  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7162
7163                                  ;;**********************************************************************
7164                                  ;*TEST 342       SBC DM1 TEST - N:C = 0101
7165                                  ;;**********************************************************************
7166  023254                          TST342:
7167  023254  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7168  023256  012700  000342                  MOV     #342,R0         ;;LOAD R0 WITH TEST NUMBER
7169  023262  013701  023306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7170  023266  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7171  023272  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
7172  023276  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
7173  023302  000257                          CCC                     ;CLEAR CODES
7174  023304  000265                          265                     ;N:C = 0101
7175
7176  023306  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7177
7178  023310  100403                          BMI     3$              ;N:C = 0010 ?
7179  023312  001402                          BEQ     3$
7180  023314  102001                          BVC     3$
7181  023316  103001                          BCC     4$
7182
7183  023320  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7184  023322  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7185  023324  001402                          BEQ     TST343          ;;BR IF YES
```

```
7186
7187  023326  011203                    MOV     (R2),R3         ;GET THE WAS DATA
7188  023330  104001            5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7189
7190                              ;;*****************************************************************
7191                              ;*TEST 343       SBC DM1 TEST - N:C = 1110
7192                              ;;*****************************************************************
7193  023332                      TST343:
7194  023332  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7195  023334  012700  000343              MOV     #343,R0         ;;LOAD R0 WITH TEST NUMBER
7196  023340  013701  023364              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7197  023344  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7198  023350  012704  000001              MOV     #1,R4           ;RESULT S / B = 1
7199  023354  012712  000001              MOV     #1,(R2)         ;[DEST] = 1
7200  023360  000257                      CCC                     ;CLEAR CODES
7201  023362  000276                      276                     ;N:C = 1110
7202
7203  023364  005612            2$:     SBC     (R2)            ;TEST THE SBC
7204
7205  023366  100403                      BMI     3$              ;N:C = 0000 ?
7206  023370  001402                      BEQ     3$
7207  023372  102401                      BVS     3$
7208  023374  103001                      BCC     4$
7209
7210  023376  104001            3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7211  023400  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
7212  023402  001402                      BEQ     TST344          ;;BR IF YES
7213
7214  023404  011203                      MOV     (R2),R3         ;GET THE WAS DATA
7215  023406  104001            5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7216
7217                              ;;*****************************************************************
7218                              ;*TEST 344       SBC DM1 TEST - N:C = 0111
7219                              ;;*****************************************************************
7220  023410                      TST344:
7221  023410  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7222  023412  012700  000344              MOV     #344,R0         ;;LOAD R0 WITH TEST NUMBER
7223  023416  013701  023440              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7224  023422  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7225  023426  012704  177777              MOV     #-1,R4          ;RESULT S / B = 177777
7226  023432  005012                      CLR     (R2)            ;[DEST] = 000000
7227  023434  000257                      CCC                     ;CLEAR CODES
7228  023436  000267                      267                     ;N:C = 0111
7229
7230  023440  005612            2$:     SBC     (R2)            ;TEST THE SBC
7231
7232  023442  100003                      BPL     3$              ;N:C = 1001 ?
7233  023444  001402                      BEQ     3$
7234  023446  102401                      BVS     3$
7235  023450  103401                      BCS     4$
7236
7237  023452  104001            3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7238  023454  020412            4$:     CMP     R4,(R2)         ;RESULT OK ?
7239  023456  001402                      BEQ     TST345          ;;BR IF YES
7240
7241  023460  011203                      MOV     (R2),R3         ;GET THE WAS DATA
```

```
7242  023462  104001                5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7243
7244                                  ;;*****************************************************************
7245                                  ;*TEST 345       NEGB - MODE 0 TEST - N:C = 0110
7246                                  ;;*****************************************************************
7247  023464                          TST345:
7248  023464  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7249  023466  012700  000345                 MOV     #345,R0         ;;LOAD R0 WITH TEST NUMBER
7250  023472  013701  023512                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7251  023476  012704  177776                 MOV     #177776,R4      ;RESULT S / B = 376 (LO BYTE)
7252  023502  012703  177402                 MOV     #177402,R3      ;[DEST] = 177402
7253  023506  000257                         CCC                     ;CLEAR FLAGS
7254  023510  000266                         266                     ;N:C = 0110
7255
7256  023512  105403                2$:     NEGB    R3              ;TEST THE NEGB
7257
7258  023514  100003                        BPL     3$              ;N:C = 1001
7259  023516  001402                        BEQ     3$
7260  023520  102401                        BVS     3$
7261  023522  103401                        BCS     4$
7262
7263  023524  104002                3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7264
7265  023526  020403                4$:     CMP     R4,R3           ;CORRECT RESULT ?
7266  023530  001401                        BEQ     TST346          ;;BR IF YES
7267
7268  023532  104002                5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7269
7270
7271                                  ;;*****************************************************************
7272                                  ;*TEST 346       NEGB - MODE 0 TEST - N:C = 0011
7273  023534                          ;;*****************************************************************
7274  023534  000004                  TST346:
7275  023536  012700  000346                 SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7276  023542  013701  023562                 MOV     #346,R0         ;;LOAD R0 WITH TEST NUMBER
7277  023546  012704  177400                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7278  023552  012703  177400                 MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7279  023556  000257                         MOV     #177400,R3      ;[DEST] = 177400
7280  023560  000263                         CCC                     ;CLEAR FLAGS
7281                                        263                     ;N:C = 0011
7282  023562  105403                2$:     NEGB    R3              ;TEST THE NEGB
7283
7284  023564  100403                        BMI     3$              ;N:C = 0100
7285  023566  001002                        BNE     3$
7286  023570  102401                        BVS     3$
7287  023572  103001                        BCC     4$
7288
7289  023574  104002                3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
7290
7291  023576  020403                4$:     CMP     R4,R3           ;CORRECT RESULT ?
7292  023600  001401                        BEQ     TST347          ;;BR IF YES
7293
7294  023602  104002                5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
7295
7296                                  ;;*****************************************************************
7297                                  ;*TEST 347       NEGB - MODE 0 TEST - N:C = 1101
```

```
                                ;;*****************************************************************
                                TST347:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #347,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177600,R4      ;RESULT S / B = 200 (LO BYTE)
                                        MOV     #177600,R3      ;[DEST] = 177600
                                        CCC                     ;CLEAR FLAGS
                                        275                     ;N:C = 1101
                                2$:     NEGB    R3              ;TEST THE NEGB
                                        BPL     3$              ;N:C = 1011
                                        BEQ     3$
                                        BVC     3$
                                        BCS     4$
                                3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
                                4$:     CMP     R4,R3           ;CORRECT RESULT ?
                                        BEQ     TST350          ;;BR IF YES
                                5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
                                ;;*****************************************************************
                                ;*TEST 350       CLRB - MODE 0 TEST - N:C = 1011
                                ;;*****************************************************************
                                TST350:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #350,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
                                        MOV     #-1,R3          ;[DEST] = 177777
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;N:C = 1011
                                2$:     CLRB    R3              ;TEST THE CLRB
                                        BMI     3$              ;N:C = 0100 ?
                                        BNE     3$
                                        BVS     3$
                                        BCC     4$
                                3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
                                4$:     CMP     R4,R3           ;RESULT CORRECT ?
                                        BEQ     TST351          ;;BR IF YES
                                5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
                                ;;*****************************************************************
                                ;*TEST 351       CLRB - MODE 0 TEST - N:C = 0100
                                ;;*****************************************************************
                                TST351:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #351,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
7354  023732  013701  023752          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7355  023736  012704  177400          MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
7356  023742  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
7357  023746  000257                  CCC                     ;CLEAR FLAGS
7358  023750  000264                  SEZ                     ;N:C = 0100
7359
7360  023752  105003          2$:     CLRB    R3              ;TEST THE CLRB
7361
7362  023754  100403                  BMI     3$              ;N:C = 0100 ?
7363  023756  001002                  BNE     3$
7364  023760  102401                  BVS     3$
7365  023762  103001                  BCC     4$
7366
7367  023764  104002          3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
7368
7369  023766  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
7370  023770  001401                  BEQ     TST352          ;;BR IF YES
7371
7372  023772  104002          5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
7373
7374                          ;;**********************************************************************
7375                          ;*TEST 352       CLRB TEST - DM2 - ODD ADDRESS
7376                          ;;**********************************************************************
7377  023774                  TST352:
7378  023774  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7379  023776  012700  000352          MOV     #352,R0         ;;LOAD R0 WITH TEST NUMBER
7380  024002  013701  024032          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7381  024006  012702  063313          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7382  024012  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
7383  024016  012705  063312          MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7384  024022  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7385  024024  012715  177777          MOV     #-1,(R5)        ;[DEST] = 177777
7386  024030  000257                  CCC                     ;SCOPE SYNC
7387
7388  024032  105023          2$:     CLRB    (R3)+           ;TEST THE CLRB
7389
7390  024034  022703  063314          CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
7391  024040  001401                  BEQ     4$              ;BR IF YES
7392
7393  024042  104005          3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7394
7395  024044  020415          4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
7396  024046  001402                  BEQ     TST353          ;;BR IF YES
7397
7398  024050  011503                  MOV     (R5),R3         ;GET THE WAS DATA
7399  024052  104001          5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7400
7401                          ;;**********************************************************************
7402                          ;*TEST 353       CLRB TEST - DM1 - ODD ADDRESS
7403                          ;;**********************************************************************
7404  024054                  TST353:
7405  024054  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7406  024056  012700  000353          MOV     #353,R0         ;;LOAD R0 WITH TEST NUMBER
7407  024062  013701  024112          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7408  024066  012702  063313          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7409  024072  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
```

```
7410  024076  012705  063312          MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7411  024102  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7412  024104  012715  177777          MOV     #-1,(R5)        ;[DEST] = 177777
7413  024110  000257                  CCC                     ;SCOPE SYNC
7414
7415  024112  105013          2$:     CLRB    (R3)            ;TEST THE CLRB
7416
7417  024114  020415                  CMP     R4,(R5)         ;CORRECT RESULT ?
7418  024116  001402                  BEQ     TST354          ;;BR IF YES
7419
7420  024120  011503                  MOV     (R5),R3         ;GET THE WAS DATA
7421  024122  104001          3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7422
7423                          ;;**********************************************************************
7424                          ;*TEST 354       CLRB TEST - DM2 - EVEN ADDRESS
7425                          ;;**********************************************************************
7426  024124                  TST354:
7427  024124  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7428  024126  012700  000354          MOV     #354,R0         ;;LOAD R0 WITH TEST NUMBER
7429  024132  013701  024156          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7430  024136  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7431  024142  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
7432  024146  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7433  024150  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7434  024154  000257                  CCC                     ;SCOPE SYNC
7435
7436  024156  105023          2$:     CLRB    (R3)+           ;TEST THE CLRB
7437
7438  024160  022703  063313          CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7439  024164  001401                  BEQ     4$              ;BR IF YES
7440
7441  024166  104005          3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
7442
7443  024170  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7444  024172  001402                  BEQ     TST355          ;;BR IF YES
7445
7446  024174  011203                  MOV     (R2),R3         ;GET THE WAS DATA
7447  024176  104001          5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7448
7449                          ;;**********************************************************************
7450                          ;*TEST 355       CLRB TEST - DM1 - EVEN ADDRESS
7451                          ;;**********************************************************************
7452  024200                  TST355:
7453  024200  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7454  024202  012700  000355          MOV     #355,R0         ;;LOAD R0 WITH TEST NUMBER
7455  024206  013701  024232          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7456  024212  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7457  024216  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
7458  024222  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7459  024224  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
7460  024230  000257                  CCC                     ;SCOPE SYNC
7461
7462  024232  105013          2$:     CLRB    (R3)            ;TEST THE CLRB
7463
7464  024234  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
7465  024236  001402                  BEQ     TST356          ;;BR IF YES
```

```
7466
7467  024240  011203                        MOV     (R2),R3         ;GET THE WAS DATA
7468  024242  104001                3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
7469
7470                                ;;*****************************************************************
7471                                ;*TEST 356       NEGB TEST - DM2 - ODD ADDRESS
7472                                ;;*****************************************************************
7473  024244                        TST356:
7474  024244  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7475  024246  012700  000356                MOV     #356,R0         ;;LOAD R0 WITH TEST NUMBER
7476  024252  013701  024302                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7477  024256  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7478  024262  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
7479  024266  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7480  024272  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7481  024274  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
7482  024300  000257                        CCC                     ;SCOPE SYNC
7483
7484  024302  105423                2$:     NEGB    (R3)+           ;TEST THE NEGB
7485
7486  024304  022703  063314                CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
7487  024310  001401                        BEQ     4$              ;BR IF YES
7488
7489  024312  104005                3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7490
7491  024314  020415                4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
7492  024316  001402                        BEQ     TST357          ;;BR IF YES
7493
7494  024320  011503                        MOV     (R5),R3         ;GET THE WAS DATA
7495  024322  104001                5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7496
7497                                ;;*****************************************************************
7498                                ;*TEST 357       NEGB TEST - DM1 - ODD ADDRESS
7499                                ;;*****************************************************************
7500  024324                        TST357:
7501  024324  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7502  024326  012700  000357                MOV     #357,R0         ;;LOAD R0 WITH TEST NUMBER
7503  024332  013701  024374                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7504                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT7
7505  024336  032737  000200  063234        BIT     #BIT7,@#BPTLOC  ;BREAKPOINT HALT SET ??
7506  024344  001401                        BEQ     .+4             ;BR IF NOT
7507  024346  000000                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
7508  024350  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
7509  024354  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
7510  024360  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
7511  024364  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7512  024366  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
7513  024372  000257                        CCC                     ;SCOPE SYNC
7514
7515  024374  105413                2$:     NEGB    (R3)            ;TEST THE NEGB
7516
7517  024376  020415                        CMP     R4,(R5)         ;CORRECT RESULT ?
7518  024400  001402                        BEQ     TST360          ;;BR IF YES
7519
7520  024402  011503                        MOV     (R5),R3         ;GET THE WAS DATA
```

```
7521  024404  104001                  3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7522
7523                                  ;;*****************************************************************
7524                                  ;*TEST 360       NEGB TEST - DM2 - EVEN ADDRESS
7525                                  ;;*****************************************************************
7526  024406                          TST360:
7527  024406  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7528  024410  012700  000360                  MOV     #360,R0         ;;LOAD R0 WITH TEST NUMBER
7529  024414  013701  024440                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7530  024420  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7531  024424  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
7532  024430  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7533  024432  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
7534  024436  000257                          CCC                     ;SCOPE SYNC
7535
7536  024440  105423                  2$:     NEGB    (R3)+           ;TEST THE NEGB
7537
7538  024442  022703  063313                  CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7539  024446  001401                          BEQ     4$              ;BR IF YES
7540
7541  024450  104005                  3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7542
7543  024452  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7544  024454  001402                          BEQ     TST361          ;;BR IF YES
7545
7546  024456  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7547  024460  104001                  5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7548
7549                                  ;;*****************************************************************
7550                                  ;*TEST 361       NEGB TEST - DM1 - EVEN ADDRESS
7551                                  ;;*****************************************************************
7552  024462                          TST361:
7553  024462  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7554  024464  012700  000361                  MOV     #361,R0         ;;LOAD R0 WITH TEST NUMBER
7555  024470  013701  024514                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7556  024474  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7557  024500  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
7558  024504  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7559  024506  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
7560  024512  000257                          CCC                     ;SCOPE SYNC
7561
7562  024514  105413                  2$:     NEGB    (R3)            ;TEST THE NEGB
7563
7564  024516  020412                          CMP     R4,(R2)         ;CORRECT RESULT ?
7565  024520  001402                          BEQ     TST362          ;;BR IF YES
7566
7567  024522  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7568  024524  104001                  3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7569
7570                                  ;;*****************************************************************
7571                                  ;*TEST 362       ADD TEST - SM0,DM0 - N:C = 1010
7572                                  ;;*****************************************************************
7573  024526                          TST362:
7574  024526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7575  024530  012700  000362                  MOV     #362,R0         ;;LOAD R0 WITH TEST NUMBER
7576  024534  013701  024556                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7577  024540  005004                    CLR     R4              ;RESULT S / B = 000000
7578  024542  012705  177777            MOV     #-1,R5          ;SRC OPR = 177777
7579  024546  012703  000001            MOV     #+1,R3          ;[DEST] = +1
7580  024552  000257                    CCC                     ;CLEAR FLAGS
7581  024554  000272                    272                     ;N:C = 1010
7582
7583  024556  060503            2$:     ADD     R5,R3           ;TEST THE ADD
7584
7585  024560  100403                    BMI     3$              ;N:C = 0101
7586  024562  001002                    BNE     3$
7587  024564  102401                    BVS     3$
7588  024566  103401                    BCS     4$
7589
7590  024570  104002            3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7591
7592  024572  020403            4$:     CMP     R4,R3           ;CORRECT RESULT ?
7593  024574  001401                    BEQ     TST363          ;;BR IF YES
7594
7595  024576  104002            5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7596
7597                            ;;**********************************************************************
7598                            ;*TEST 363       ADD TEST - SM0,DM0 - N:C = 0101
7599                            ;;**********************************************************************
7600  024600                    TST363:
7601  024600  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7602  024602  012700  000363            MOV     #363,R0         ;;LOAD R0 WITH TEST NUMBER
7603  024606  013701  024632            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7604  024612  012704  100006            MOV     #100006,R4      ;RESULT S / B = 100006
7605  024616  012705  077777            MOV     #77777,R5       ;SRC OPR = 77777
7606  024622  012703  000007            MOV     #7,R3           ;[DEST] = 7
7607  024626  000257                    CCC                     ;CLEAR FLAGS
7608  024630  000265                    265                     ;N:C = 0101
7609
7610  024632  060503            2$:     ADD     R5,R3           ;TEST THE ADD
7611
7612  024634  100003                    BPL     3$              ;N:C = 1010
7613  024636  001402                    BEQ     3$
7614  024640  102001                    BVC     3$
7615  024642  103001                    BCC     4$
7616
7617  024644  104002            3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7618
7619  024646  020403            4$:     CMP     R4,R3           ;CORRECT RESULT ?
7620  024650  001401                    BEQ     TST364          ;;BR IF YES
7621
7622  024652  104002            5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7623
7624                            ;;**********************************************************************
7625                            ;*TEST 364       ADD SM1,DM0 TEST
7626                            ;;**********************************************************************
7627  024654                    TST364:
7628  024654  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7629  024656  012700  000364            MOV     #364,R0         ;;LOAD R0 WITH TEST NUMBER
7630  024662  013701  024702            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7631  024666  012704  063322            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7632  024672  012705  063276            MOV     #ATA,R5         ;SOURCE ADDR = ATA
```

```
7633  024676  005003                  CLR     R3              ;[DEST] = 0
7634  024700  000257                  CCC                     ;SCOPE SYNC
7635
7636  024702  061503          2$:     ADD     (R5),R3         ;TEST THE ADD - SM1,DMO
7637
7638  024704  020403                  CMP     R4,R3           ;RESULT = #DWTA?
7639  024706  001401                  BEQ     4$              ;BR IF YES
7640
7641  024710  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7642
7643  024712  022705  063276  4$:     CMP     #ATA,R5         ;DID ADD CHANGE REG.
7644  024716  001401                  BEQ     TST365          ;;BR IF NOT
7645
7646  024720  104005          5$:     ERROR   5               ;REG GOT MODIFIED
7647
7648                          ;;**********************************************************************
7649                          ;*TEST 365       ADD SM2,DMO TEST
7650                          ;;**********************************************************************
7651  024722                  TST365:
7652  024722  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7653  024724  012700  000365          MOV     #365,R0         ;;LOAD R0 WITH TEST NUMBER
7654  024730  013701  024750          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7655  024734  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7656  024740  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7657  024744  005003                  CLR     R3              ;[DEST] = 0
7658  024746  000257                  CCC                     ;SCOPE SYNC
7659
7660  024750  062503          2$:     ADD     (R5)+,R3        ;TEST THE ADD - SM2,DMO
7661
7662  024752  020403                  CMP     R4,R3           ;RESULT = #DWTA
7663  024754  001401                  BEQ     4$              ;BR IF YES
7664
7665  024756  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7666
7667  024760  022705  063300  4$:     CMP     #ATA+2,R5       ;DID ADD AUTO INCREMENT SOURCE REG?
7668  024764  001401                  BEQ     TST366          ;;BR IF YES
7669
7670  024766  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7671
7672                          ;;**********************************************************************
7673                          ;*TEST 366       ADD SM3,DMO TEST
7674                          ;;**********************************************************************
7675  024770                  TST366:
7676  024770  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7677  024772  012700  000366          MOV     #366,R0         ;;LOAD R0 WITH TEST NUMBER
7678  024776  013701  025022          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7679  025002  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7680  025006  012705  063306          MOV     #ATA+10,R5      ;R5 POINTS TO SOURCE ADDR
7681  025012  010437  063312          MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7682  025016  005003                  CLR     R3              ;[DEST] = 0
7683  025020  000257                  CCC                     ;SCOPE SYNC
7684
7685  025022  063503          2$:     ADD     @(R5)+,R3       ;TEST THE ADD - SM3,DMO
7686
7687  025024  020437  063312          CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7688  025030  001401                  BEQ     4$              ;BR IF YES
```

```
7689
7690  025032  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7691
7692  025034  022705  063310        4$:     CMP     #ATA+12,R5      ;DID ADD AUTO INCREMENT SOURCE REG?
7693  025040  001401                        BEQ     TST367          ;;BR IF YES
7694
7695  025042  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7696
7697                                ;;**************************************************************
7698                                ;*TEST 367       ADD SM4,DM0 TEST
7699                                ;;**************************************************************
7700  025044                        TST367:
7701  025044  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7702  025046  012700  000367               MOV     #367,R0         ;;LOAD R0 WITH TEST NUMBER
7703  025052  013701  025072               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7704  025056  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7705  025062  012705  063300               MOV     #ATA+2,R5       ;SOURCE ADDR = ATA
7706  025066  005003                       CLR     R3              ;[DEST] = 0
7707  025070  000257                       CCC                     ;SCOPE SYNC
7708
7709  025072  064503                2$:     ADD     -(R5),R3        ;TEST THE ADD - SM4,DM0
7710
7711  025074  020403                       CMP     R4,R3           ;RESULT = #DWTA?
7712  025076  001401                       BEQ     4$              ;BR IF YES
7713
7714  025100  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7715
7716  025102  022705  063276        4$:     CMP     #ATA,R5         ;DID SOURCE REG GET DECREMENTED?
7717  025106  001401                        BEQ     TST370          ;;BR IF YES
7718
7719  025110  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG
7720
7721                                ;;**************************************************************
7722                                ;*TEST 370       ADD SM5,DM0 TEST
7723                                ;;**************************************************************
7724  025112                        TST370:
7725  025112  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7726  025114  012700  000370               MOV     #370,R0         ;;LOAD R0 WITH TEST NUMBER
7727  025120  013701  025144               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7728  025124  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7729  025130  012705  063310               MOV     #ATA+12,R5      ;R5 POINTS TO SOURCE ADDR
7730  025134  010437  063312               MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7731  025140  005003                       CLR     R3              ;[DEST] = 0
7732  025142  000257                       CCC                     ;SCOPE SYNC
7733
7734  025144  065503                2$:     ADD     @-(R5),R3       ;TEST THE ADD - SM5,DM0
7735
7736  025146  020437  063312               CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7737  025152  001401                       BEQ     4$              ;BR IF YES
7738
7739  025154  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7740
7741  025156  022705  063306        4$:     CMP     #ATA+10,R5      ;DID ADD DECREMENT SOURCE REG?
7742  025162  001401                        BEQ     TST371          ;;BR IF YES
7743
7744  025164  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
```

```
7745
7746                                    ;;*****************************************************************
7747                                    ;*TEST 371       ADD SM6,DM0 TEST
7748                                    ;;*****************************************************************
7749  025166                            TST371:
7750  025166  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7751  025170  012700  000371                   MOV     #371,R0         ;;LOAD R0 WITH TEST NUMBER
7752  025174  013701  025214                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7753  025200  012704  063312                   MOV     #MBUF0,R4       ;RESULT S / B = MBUF0
7754  025204  012705  063276                   MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7755  025210  005003                           CLR     R3              ;[DEST] = 0
7756  025212  000257                           CCC                     ;SCOPE SYNC
7757
7758  025214  066503  000010           2$:     ADD     10(R5),R3       ;TEST THE ADD - SM6,DM0
7759
7760  025220  020403                           CMP     R4,R3           ;RESULT =MBUF0?
7761  025222  001401                           BEQ     TST372          ;;BR IF YES
7762
7763  025224  104002                   3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7764
7765                                    ;;*****************************************************************
7766                                    ;*TEST 372       ADD SM7,DM0 TEST
7767                                    ;;*****************************************************************
7768  025226                            TST372:
7769  025226  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7770  025230  012700  000372                   MOV     #372,R0         ;;LOAD R0 WITH TEST NUMBER
7771  025234  013701  025260                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7772  025240  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7773  025244  012705  063276                   MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7774  025250  010437  063312                   MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7775  025254  005003                           CLR     R3              ;[DEST] = 0
7776  025256  000257                           CCC                     ;SCOPE SYNC
7777
7778  025260  067503  000010           2$:     ADD     @10(R5),R3      ;TEST THE ADD - SM7,DM0
7779
7780  025264  020403                           CMP     R4,R3           ;RESULT = #DWTA?
7781  025266  001401                           BEQ     TST373          ;;BR IF YES
7782
7783  025270  104002                   3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7784
7785                                    ;;*****************************************************************
7786                                    ;*TEST 373       ADD SM1,DM1 TEST
7787                                    ;;*****************************************************************
7788  025272                            TST373:
7789  025272  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7790  025274  012700  000373                   MOV     #373,R0         ;;LOAD R0 WITH TEST NUMBER
7791  025300  013701  025324                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7792  025304  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7793  025310  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7794  025314  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
7795  025320  005012                           CLR     (R2)            ;[DEST] = 0
7796  025322  000257                           CCC                     ;SCOPE SYNC
7797
7798  025324  061512                   2$:     ADD     (R5),(R2)       ;TEST THE ADD - SM1,DM1
7799
7800  025326  020412                           CMP     R4,(R2)         ;RESULT = #DWTA?
```

```
                                        BEQ     TST374          ;;BR IF YES

                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 374       ADD SM2,DM1 TEST
                                ;;*****************************************************************
                                TST374:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #374,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                        MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                        CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC

                                2$:     ADD     (R5)+,(R2)      ;TEST THE ADD - SM2,DM1

                                        CMP     R4,(R2)         ;RESULT = #DWTA?
                                        BEQ     TST375          ;;BR IF YES

                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 375       ADD SM1,DM2 TEST
                                ;;*****************************************************************
                                TST375:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #375,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #DWTA,R4        ;RESULT S / B = #DWTA
                                        MOV     #ATA,R5         ;SOURCE ADDR = ATA
                                        MOV     R2,R3           ;[R3] = DEST ADDR
                                        CLR     (R2)            ;[DEST] = 0
                                        CCC                     ;SCOPE SYNC

                                2$:     ADD     (R5),(R3)+      ;TEST THE ADD - SM1,DM2

                                        CMP     R4,(R2)         ;RESULT = #DWTA?
                                        BEQ     4$              ;BR IF YES

                                        MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
                                        MOV     (R2),R3         ;GET WAS DATA
                                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

                                        MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
                                4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG
                                        BEQ     TST376          ;;BR IF YES

                                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG

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

```
7857
7858                                    ;*TEST 376      ADD SM2,DM2 TEST
7859  025470                            ;;**************************************************************************
7860  025470  000004                    TST376:
7861  025472  012700  000376                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7862  025476  013701  025524                   MOV     #376,R0         ;;LOAD R0 WITH TEST NUMBER
7863  025502  012702  063312                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7864  025506  012704  063322                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7865  025512  012705  063276                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7866  025516  010203                           MOV     #ATA,R5         ;SOURCE ADDR = ATA
7867  025520  005012                           MOV     R2,R3           ;[R3] = DEST ADDR
7868  025522  000257                           CLR     (R2)            ;[DEST] = 0
7869                                           CCC                     ;SCOPE SYNC
7870  025524  062523                   2$:     ADD     (R5)+,(R3)+     ;TEST THE ADD - SM2,DM2
7871
7872  025526  020412                           CMP     R4,(R2)         ;RESULT = #DWTA
7873  025530  001406                           BEQ     4$              ;BR IF YES
7874
7875  025532  010337  063316                   MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
7876  025536  011203                           MOV     (R2),R3         ;GET WAS DATA
7877  025540  104001                   3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7878
7879  025542  013703  063316                   MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
7880  025546  022703  063314           4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG?
7881  025552  001401                           BEQ     TST377          ;;BR IF YES
7882
7883  025554  104005                   5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7884
7885                                    ;;**************************************************************************
7886                                    ;*TEST 377      ADD SM1,DM3 TEST
7887                                    ;;**************************************************************************
7888  025556                            TST377:
7889  025556  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7890  025560  012700  000377                   MOV     #377,R0         ;;LOAD R0 WITH TEST NUMBER
7891  025564  013701  025614                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7892  025570  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7893  025574  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7894  025600  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
7895  025604  012703  063306                   MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
7896  025610  005012                           CLR     (R2)            ;[DEST] = 0
7897  025612  000257                           CCC                     ;SCOPE SYNC
7898
7899  025614  061533                   2$:     ADD     (R5),@(R3)+     ;TEST THE ADD - SM1,DM3
7900
7901  025616  020412                           CMP     R4,(R2)         ;RESULT = #DWTA?
7902  025620  001406                           BEQ     4$              ;BR IF YES
7903
7904  025622  010337  063316                   MOV     R3,@#MBUF1      ;SAVE R3
7905  025626  011203                           MOV     (R2),R3         ;GET WAS DATA
7906  025630  104001                   3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7907
7908  025632  013703  063316                   MOV     @#MBUF1,R3      ;RESTORE R3
7909  025636  022703  063310           4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
7910  025642  001401                           BEQ     TST400          ;;BR IF YES
7911
7912  025644  104005                   5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
```

```
                                ;;*****************************************************************
                                ;*TEST 400       ADD SM2,DM3 TEST
                                ;;*****************************************************************
025646                          TST400:
025646  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025650  012700  000400                  MOV     #400,R0         ;;LOAD R0 WITH TEST NUMBER
025654  013701  025704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025660  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025664  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025670  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025674  012703  063306                  MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
025700  005012                          CLR     (R2)            ;[DEST] = 0
025702  000257                          CCC                     ;SCOPE SYNC
                                
025704  062533                  2$:     ADD     (R5)+,@(R3)+    ;TEST THE ADD - SM2,DM3
                                
025706  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
025710  001406                          BEQ     4$              ;BR IF YES
                                
025712  010337  063316                  MOV     R3,@#MBUF1      ;SAVE R3
025716  011203                          MOV     (R2),R3         ;GET WAS DATA
025720  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
025722  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE R3
025726  022703  063310          4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
025732  001401                          BEQ     TST401          ;;BR IF YES
                                
025734  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
                                
                                ;;*****************************************************************
                                ;*TEST 401       ADD SM1,DM4 TEST
                                ;;*****************************************************************
025736                          TST401:
025736  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025740  012700  000401                  MOV     #401,R0         ;;LOAD R0 WITH TEST NUMBER
025744  013701  025774                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025750  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025754  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025760  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025764  012703  063314                  MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
025770  005012                          CLR     (R2)            ;[DEST] = 0
025772  000257                          CCC                     ;SCOPE SYNC
                                
025774  061543                  2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM1,DM4
                                
026000  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
026002  001406                          BEQ     4$              ;BR IF YES
                                
026004  010337  063316                  MOV     R3,@#MBUF1      ;SAVE R3
026010  011203                          MOV     (R2),R3         ;GET WAS DATA
026012  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
026014  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE R3
026020  020302                  4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
026022  001401                          BEQ     TST402          ;;BR IF YES
```

```
7969
7970  026022  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
7971
7972                                ;;**********************************************************************
7973                                ;*TEST 402       ADD SM2,DM4 TEST
7974                                ;;**********************************************************************
7975  026024                        TST402:
7976  026024  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7977  026026  012700  000402               MOV     #402,R0         ;;LOAD R0 WITH TEST NUMBER
7978  026032  013701  026062               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7979  026036  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7980  026042  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7981  026046  012705  063276               MOV     #ATA,R5         ;SOURCE ADDR = ATA
7982  026052  012703  063314               MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
7983  026056  005012                       CLR     (R2)            ;[DEST] = 0
7984  026060  000257                       CCC                     ;SCOPE SYNC
7985
7986  026062  061543                2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM2,DM4
7987
7988  026064  020412                       CMP     R4,(R2)         ;RESULT = #DWTA?
7989  026066  001406                       BEQ     4$              ;BR IF YES
7990
7991  026070  010337  063316               MOV     R3,@#MBUF1      ;SAVE R3
7992  026074  011203                       MOV     (R2),R3         ;GET WAS DATA
7993  026076  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7994
7995  026100  013703  063316               MOV     @#MBUF1,R3      ;RESTORE R3
7996  026104  020302                4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
7997  026106  001401                       BEQ     TST403          ;;BR IF YES
7998
7999  026110  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
8000
8001                                ;;**********************************************************************
8002                                ;*TEST 403       ADD SM1,DM5 TEST
8003                                ;;**********************************************************************
8004  026112                        TST403:
8005  026112  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8006  026114  012700  000403               MOV     #403,R0         ;;LOAD R0 WITH TEST NUMBER
8007  026120  013701  026150               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8008  026124  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8009  026130  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8010  026134  012705  063276               MOV     #ATA,R5         ;SOURCE ADDR = ATA
8011  026140  012703  063310               MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
8012  026144  005012                       CLR     (R2)            ;[DEST] = 0
8013  026146  000257                       CCC                     ;SCOPE SYNC
8014
8015  026150  061553                2$:     ADD     (R5),@-(R3)     ;TEST THE ADD - SM1,DM5
8016
8017  026152  020412                       CMP     R4,(R2)         ;RESULT = #DWTA?
8018  026154  001406                       BEQ     4$              ;BR IF YES
8019
8020  026156  010337  063316               MOV     R3,@#MBUF1      ;SAVE R3
8021  026162  011203                       MOV     (R2),R3         ;GET WAS DATA
8022  026164  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8023
8024  026166  013703  063316               MOV     @#MBUF1,R3      ;RESTORE R3
```

```
8025  026172  022703  063306        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8026  026176  001401                        BEQ     TST404          ;;BR IF YES
8027
8028  026200  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
8029
8030                                  ;;*******************************************************************
8031                                  ;*TEST 404       ADD SM2,DM5 TEST
8032                                  ;;*******************************************************************
8033  026202                          TST404:
8034  026202  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8035  026204  012700  000404                MOV     #404,R0         ;;LOAD R0 WITH TEST NUMBER
8036  026210  013701  026240                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8037  026214  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8038  026220  012704  063322                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8039  026224  012705  063276                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8040  026230  012703  063310                MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
8041  026234  005012                        CLR     (R2)            ;[DEST] = 0
8042  026236  000257                        CCC                     ;SCOPE SYNC
8043
8044  026240  062553                2$:     ADD     (R5)+,@-(R3)    ;TEST THE ADD - SM2,DM5
8045
8046  026242  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8047  026244  001406                        BEQ     4$              ;BR IF YES
8048
8049  026246  010337  063316                MOV     R3,@#MBUF1      ;SAVE R3
8050  026252  011203                        MOV     (R2),R3         ;GET WAS DATA
8051  026254  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8052
8053  026256  013703  063316                MOV     @#MBUF1,R3      ;RESTORE R3
8054  026262  022703  063306        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8055  026266  001401                        BEQ     TST405          ;;BR IF YES
8056
8057  026270  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
8058
8059                                  ;;*******************************************************************
8060                                  ;*TEST 405       ADD SM1,DM6 TEST
8061                                  ;;*******************************************************************
8062  026272                          TST405:
8063  026272  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8064  026274  012700  000405                MOV     #405,R0         ;;LOAD R0 WITH TEST NUMBER
8065  026300  013701  026330                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8066  026304  012702  063316                MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
8067  026310  012704  063322                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8068  026314  012705  063276                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8069  026320  012703  063312                MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
8070  026324  005012                        CLR     (R2)            ;[DEST] = 0
8071  026326  000257                        CCC                     ;SCOPE SYNC
8072
8073  026330  061563  000004        2$:     ADD     (R5),4(R3)      ;TEST THE ADD - SM1,DM6
8074
8075  026334  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8076  026336  001402                        BEQ     TST406          ;;BR IF YES
8077
8078  026340  011203                        MOV     (R2),R3         ;GET WAS DATA
8079  026342  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8080
```

```
                                ;;*****************************************************************
                                ;*TEST 406       ADD SM2,DM6 TEST
                                ;;*****************************************************************
026344                          TST406:
026344  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026346  012700  000406                   MOV     #406,R0         ;;LOAD R0 WITH TEST NUMBER
026352  013701  026402                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026356  012702  063316                   MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
026362  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026366  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
026372  012703  063312                   MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
026376  005012                          CLR     (R2)            ;[DEST] = 0
026400  000257                          CCC                     ;SCOPE SYNC
                                
026402  062563  000004          2$:     ADD     (R5)+,4(R3)     ;TEST THE ADD - SM2,DM6
                                
026406  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
026410  001402                          BEQ     TST407          ;;BR IF YES
                                
026412  011203                          MOV     (R2),R3         ;GET WAS DATA
026414  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 407       ADD SM1,DM7 TEST
                                ;;*****************************************************************
026416                          TST407:
026416  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026420  012700  000407                   MOV     #407,R0         ;;LOAD R0 WITH TEST NUMBER
026424  013701  026452                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026430  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026434  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026440  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
026444  010503                          MOV     R5,R3           ;BASE DEST ADDR = ATA
026446  005012                          CLR     (R2)            ;[DEST] = 0
026450  000257                          CCC                     ;SCOPE SYNC
                                
026452  061573  000010          2$:     ADD     (R5),@10(R3)    ;TEST THE ADD - SM1,DM7
                                
026456  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
026460  001402                          BEQ     TST410          ;;BR IF YES
                                
026462  011203                          MOV     (R2),R3         ;GET WAS DATA
026464  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 410       ADD SM2,DM7 TEST
                                ;;*****************************************************************
026466                          TST410:
026466  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026470  012700  000410                   MOV     #410,R0         ;;LOAD R0 WITH TEST NUMBER
026474  013701  026522                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026500  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026504  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026510  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
026514  010503                          MOV     R5,R3           ;BASE DEST ADDR = ATA
026516  005012                          CLR     (R2)            ;[DEST] = 0
```

```
8137  026520  000257                  CCC                     ;SCOPE SYNC
8138
8139  026522  062573  000010  2$:     ADD     (R5)+,@10(R3)   ;TEST THE ADD - SM2,DM7
8140
8141  026526  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
8142  026530  001402                  BEQ     TST411          ;;BR IF YES
8143
8144  026532  011203                  MOV     (R2),R3         ;GET WAS DATA
8145  026534  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8146
8147                          ;;**********************************************************************
8148                          ;*TEST 411      "XOR RA,RB" TEST - A=B=000000 N:C=1010
8149                          ;;**********************************************************************
8150  026536                  TST411:
8151  026536  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8152  026540  012700  000411          MOV     #411,R0         ;;LOAD R0 WITH TEST NUMBER
8153  026544  013701  026560          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8154  026550  005004                  CLR     R4              ;RESULT AND MASK = 000000
8155  026552  005003                  CLR     R3              ;[DEST] = 000000
8156  026554  000257                  CCC                     ;SCOPE SYNC
8157  026556  000272                  272                     ;MAKE N:C=1010
8158
8159  026560  074403          2$:     XOR     R4,R3           ;TEST THE XOR
8160
8161  026562  100403                  BMI     3$              ;N:C=0100 ??
8162  026564  001002                  BNE     3$
8163  026566  102401                  BVS     3$
8164  026570  103001                  BCC     4$
8165
8166  026572  104002          3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8167
8168  026574  020403          4$:     CMP     R4,R3           ;RESULT CORRECT?
8169  026576  001401                  BEQ     TST412          ;;BR IF YES
8170
8171  026600  104002          5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8172
8173                          ;;**********************************************************************
8174                          ;*TEST 412      "XOR RA,RB" TEST - A=B=177777 N:C=0101
8175                          ;;**********************************************************************
8176  026602                  TST412:
8177  026602  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8178  026604  012700  000412          MOV     #412,R0         ;;LOAD R0 WITH TEST NUMBER
8179  026610  013701  026630          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8180  026614  005004                  CLR     R4              ;RESULT  = 000000
8181  026616  012705  177777          MOV     #-1,R5          ;MASK = 177777
8182  026622  010503                  MOV     R5,R3           ;[DEST]=177777
8183  026624  000257                  CCC                     ;SCOPE SYNC
8184  026626  000265                  265                     ;MAKE N:C=0101
8185
8186  026630  074503          2$:     XOR     R5,R3           ;TEST THE XOR
8187
8188  026632  100403                  BMI     3$              ;N:C=0101 ??
8189  026634  001002                  BNE     3$
8190  026636  102401                  BVS     3$
8191  026640  103401                  BCS     4$
8192
```

```
8193  026642  104002                3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8194
8195  026644  020403                4$:     CMP     R4,R3           ;RESULT CORRECT?
8196  026646  001401                        BEQ     TST413          ;;BR IF YES
8197
8198  026650  104002                5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8199
8200                                ;;**********************************************************************
8201                                ;*TEST 413       'XOR RA,RB'' TEST - A=125252,B=052525 N:C=0110
8202                                ;;**********************************************************************
8203  026652                        TST413:
8204  026652  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8205  026654  012700  000413                MOV     #413,R0         ;;LOAD R0 WITH TEST NUMBER
8206  026660  013701  026704                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8207  026664  012704  177777                MOV     #-1,R4          ;RESULT S/B = 177777
8208  026670  012705  125252                MOV     #125252,R5      ;MASK=125252
8209  026674  012703  052525                MOV     #052525,R3      ;[DEST] = 052525
8210  026700  000257                        CCC                     ;SCOPE SYNC
8211  026702  000266                        266                     ;MAKE N:C=0110
8212
8213  026704  074503                2$:     XOR     R5,R3           ;TEST THE XOR
8214
8215  026706  100003                        BPL     3$              ;N:C=1000 ??
8216  026710  001402                        BEQ     3$
8217  026712  102401                        BVS     3$
8218  026714  103001                        BCC     4$
8219
8220  026716  104002                3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8221
8222  026720  020403                4$:     CMP     R4,R3           ;RESULT CORRECT?
8223  026722  001401                        BEQ     TST414          ;;BR IF YES
8224
8225  026724  104002                5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8226
8227                                ;;**********************************************************************
8228                                ;*TEST 414       'XOR RA,RB'' TEST - A=052525,B=125252 N:C=1001
8229                                ;;**********************************************************************
8230  026726                        TST414:
8231  026726  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8232  026730  012700  000414                MOV     #414,R0         ;;LOAD R0 WITH TEST NUMBER
8233  026734  013701  026760                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8234  026740  012704  177777                MOV     #-1,R4          ;RESULT S/B = 177777
8235  026744  012705  052525                MOV     #52525,R5       ;MASK=052525
8236  026750  012703  125252                MOV     #125252,R3      ;[DEST] = 125252
8237  026754  000257                        CCC                     ;SCOPE SYNC
8238  026756  000271                        271                     ;MAKE N:C=1001
8239
8240  026760  074503                2$:     XOR     R5,R3           ;TEST THE XOR
8241
8242  026762  100003                        BPL     3$              ;N:C=1001 ??
8243  026764  001402                        BEQ     3$
8244  026766  102401                        BVS     3$
8245  026770  103401                        BCS     4$
8246
8247  026772  104002                3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8248
```

```
8249  026774  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
8250  026776  001401                          BEQ     TST415          ;;BR IF YES
8251
8252  027000  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8253
8254                                  ;;**********************************************************************
8255                                  ;*TEST 415       'XOR RA,(RB)'' TEST - A=B=000000 N:C=1010
8256                                  ;;**********************************************************************
8257  027002                          TST415:
8258  027002  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8259  027004  012700  000415                  MOV     #415,R0         ;;LOAD R0 WITH TEST NUMBER
8260  027010  013701  027032                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8261  027014  005004                          CLR     R4              ;RESULT S / B = 000000
8262  027016  005005                          CLR     R5              ;MASK = 000000
8263  027020  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8264  027024  005012                          CLR     (R2)            ;[DEST] = 000000
8265  027026  000257                          CCC                     ;SCOPE SYNC
8266  027030  000272                          272                     ;MAKE N:C=1010
8267
8268  027032  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8269
8270  027034  100403                          BMI     3$              ;N:C = 0100 ??
8271  027036  001002                          BNE     3$
8272  027040  102401                          BVS     3$
8273  027042  103001                          BCC     4$
8274
8275  027044  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8276
8277  027046  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8278  027050  001402                          BEQ     TST416          ;;BR IF YES
8279
8280  027052  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8281  027054  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8282
8283                                  ;;**********************************************************************
8284                                  ;*TEST 416       'XOR RA,(RB)'' TEST - A=B=177777 N:C=0101
8285                                  ;;**********************************************************************
8286  027056                          TST416:
8287  027056  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8288  027060  012700  000416                  MOV     #416,R0         ;;LOAD R0 WITH TEST NUMBER
8289  027064  013701  027112                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8290  027070  005004                          CLR     R4              ;RESULT S / B = 000000
8291  027072  012705  177777                  MOV     #-1,R5          ;MASK = 177777
8292  027076  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8293  027102  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
8294  027106  000257                          CCC                     ;SCOPE SYNC
8295  027110  000265                          265                     ;MAKE N:C=0101
8296
8297  027112  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8298
8299  027114  100403                          BMI     3$              ;N:C = 0101 ??
8300  027116  001002                          BNE     3$
8301  027120  102401                          BVS     3$
8302  027122  103401                          BCS     4$
8303
8304  027124  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
```

```
8305
8306  027126  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8307  027130  001402                          BEQ     TST417          ;;BR IF YES
8308
8309  027132  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8310  027134  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8311
8312                                  ;;**********************************************************************
8313                                  ;*TEST 417      'XOR RA,(RB)'' TEST - A=125252,B=052525 N:C=0110
8314                                  ;;**********************************************************************
8315  027136                          TST417:
8316  027136  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8317  027140  012700  000417                  MOV     #417,R0         ;;LOAD R0 WITH TEST NUMBER
8318  027144  013701  027206                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8319                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT8
8320  027150  032737  000400  063234          BIT     #BIT8,@#BPTLOC  ;BREAKPOINT HALT SET ??
8321  027156  001401                          BEQ     .+4             ;BR IF NOT
8322  027160  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
8323  027162  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8324  027166  012705  125252                  MOV     #125252,R5      ;MASK = 125252
8325  027172  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8326  027176  012712  052525                  MOV     #052525,(R2)    ;[DEST] = 052525
8327  027202  000257                          CCC                     ;SCOPE SYNC
8328  027204  000266                          266                     ;MAKE N:C=0110
8329
8330  027206  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8331
8332  027210  100003                          BPL     3$              ;N:C = 1000 ??
8333  027212  001402                          BEQ     3$
8334  027214  102401                          BVS     3$
8335  027216  103001                          BCC     4$
8336
8337  027220  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8338
8339  027222  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8340  027224  001402                          BEQ     TST420          ;;BR IF YES
8341
8342  027226  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8343  027230  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8344
8345                                  ;;**********************************************************************
8346                                  ;*TEST 420      'XOR RA,(RB)'' TEST - A=052525,B=125252 N:C=1001
8347                                  ;;**********************************************************************
8348  027232                          TST420:
8349  027232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8350  027234  012700  000420                  MOV     #420,R0         ;;LOAD R0 WITH TEST NUMBER
8351  027240  013701  027270                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8352  027244  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8353  027250  012705  052525                  MOV     #052525,R5      ;MASK = 052525
8354  027254  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8355  027260  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
8356  027264  000257                          CCC                     ;SCOPE SYNC
8357  027266  000271                          271                     ;MAKE N:C=1001
8358
8359  027270  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8360
```

```
8361  027272  100003                  BPL     3$              ;N:C = 1001 ??
8362  027274  001402                  BEQ     3$
8363  027276  102401                  BVS     3$
8364  027300  103401                  BCS     4$
8365
8366  027302  104001          3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8367
8368  027304  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8369  027306  001402                  BEQ     TST421          ;;BR IF YES
8370
8371  027310  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8372  027312  104001          5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8373
8374                          ;;**************************************************************
8375                          ;*TEST 421       SUB TEST SM0,DM0 - (SRC) = (DEST) = +,+
8376                          ;;**************************************************************
8377  027314                  TST421:
8378  027314  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8379  027316  012700  000421          MOV     #421,R0         ;;LOAD R0 WITH TEST NUMBER
8380  027322  013701  027342          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8381  027326  005004                  CLR     R4              ;RESULT S / B = 0
8382  027330  012703  052525          MOV     #052525,R3      ;[R3] = DEST OP = 52525
8383  027334  010305                  MOV     R3,R5           ;[R5] = SRC OP = 52525
8384  027336  000257                  CCC                     ;CLEAR FLAGS
8385  027340  000273                  273                     ;MAKE N:C = 1011
8386
8387  027342  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8388
8389  027344  100403                  BMI     3$
8390  027346  001002                  BNE     3$              ;DID N:C = 0100
8391  027350  102401                  BVS     3$
8392  027352  103001                  BCC     4$
8393
8394  027354  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8395
8396  027356  020304          4$:     CMP     R3,R4           ;WAS RESULT = 0?
8397  027360  001401                  BEQ     TST422          ;;BR IF YES
8398
8399  027362  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8400
8401                          ;;**************************************************************
8402                          ;*TEST 422       SUB TEST SM0,DM0 - (SRC) = (DEST) = -,-
8403                          ;;**************************************************************
8404  027364                  TST422:
8405  027364  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8406  027366  012700  000422          MOV     #422,R0         ;;LOAD R0 WITH TEST NUMBER
8407  027372  013701  027412          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8408  027376  005004                  CLR     R4              ;RESULT S / B = 0
8409  027400  012703  125252          MOV     #125252,R3      ;[R3] = DEST OP = 125252
8410  027404  010305                  MOV     R3,R5           ;[R5] = SOURCE OP = 125252
8411  027406  000257                  CCC                     ;CLEAR FLAGS
8412  027410  000273                  273                     ;MAKE N:C = 1011
8413
8414  027412  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8415
8416  027414  100403                  BMI     3$
```

```
8417  027416  001002                  BNE     3$              ;N:C = 0100?
8418  027420  102401                  BVS     3$
8419  027422  103001                  BCC     4$
8420
8421  027424  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8422
8423  027426  020304          4$:     CMP     R3,R4           ;RESULT = 0?
8424  027430  001401                  BEQ     TST423          ;;BR IF YES
8425
8426  027432  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8427
8428                          ;;**********************************************************************
8429                          ;*TEST 423       SUB TEST SM0,DM0 - (SRC) = (DEST) = -,+
8430                          ;;**********************************************************************
8431  027434                  TST423:
8432  027434  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8433  027436  012700  000423          MOV     #423,R0         ;;LOAD R0 WITH TEST NUMBER
8434  027442  013701  027466          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8435  027446  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
8436  027452  012703  000001          MOV     #1,R3           ;[R3] = DEST OP = 1
8437  027456  012705  177777          MOV     #-1,R5          ;[R5] = SRC OP = -1
8438  027462  000257                  CCC                     ;CLEAR FLAGS
8439  027464  000276                  276                     ;MAKE N:C = 1110
8440
8441  027466  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8442
8443  027470  100403                  BMI     3$
8444  027472  001402                  BEQ     3$              ;N:C = 0001
8445  027474  102401                  BVS     3$
8446  027476  103401                  BCS     4$
8447
8448  027500  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8449
8450  027502  020304          4$:     CMP     R3,R4           ;RESULT = +2?
8451  027504  001401                  BEQ     TST424          ;;BR IF YES
8452
8453  027506  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8454
8455                          ;;**********************************************************************
8456                          ;*TEST 424       SUB TEST SM0,DM0 (SRC) = -(DEST) = +,-
8457                          ;;**********************************************************************
8458  027510                  TST424:
8459  027510  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8460  027512  012700  000424          MOV     #424,R0         ;;LOAD R0 WITH TEST NUMBER
8461  027516  013701  027542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8462  027522  012704  177776          MOV     #-2,R4          ;RESULT S / B = -2
8463  027526  012703  177777          MOV     #-1,R3          ;[R3] = [DEST] = -1
8464  027532  012705  000001          MOV     #1,R5           ;[R5] = [SOURCE] = +1
8465  027536  000257                  CCC                     ;CLEAR FLAGS
8466  027540  000267                  267                     ;MAKE N:C = 0111
8467
8468  027542  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8469
8470  027544  100003                  BPL     3$
8471  027546  001402                  BEQ     3$              ;N:C = 1000
8472  027550  102401                  BVS     3$
```

```
8473  027552  103001                  BCC     4$
8474
8475  027554  104002          3$:     ERROR   2               ;SUB DID NOT ALTER CODES PROPERLY
8476
8477  027556  020403          4$:     CMP     R4,R3           ;RESULT = -2?
8478  027560  001401                  BEQ     TST425          ;;BR IF YES
8479
8480  027562  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8481
8482                            ;;*******************************************************************
8483                            ;*TEST 425      SUB TEST SM0,DM0 - 'V' BIT SETS
8484                            ;;*******************************************************************
8485  027564                    TST425:
8486  027564  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8487  027566  012700  000425           MOV     #425,R0         ;;LOAD R0 WITH TEST NUMBER
8488  027572  013701  027616           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8489  027576  012704  077777           MOV     #77777,R4       ;RESULT = 77777
8490  027602  012703  100000           MOV     #100000,R3      ;[R3] = DEST OP = 100000
8491  027606  012705  000001           MOV     #1,R5           ;[R5] = SRC OP = 1
8492  027612  000257                   CCC                     ;CLEAR FLAGS
8493  027614  000274                   274                     ;MAKE N:C = 1100
8494
8495  027616  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8496
8497  027620  100403                  BMI     3$
8498  027622  001402                  BEQ     3$              ;N:C = 0011 ('V' BIT SHOULD SET)
8499  027624  102001                  BVC     3$
8500  027626  103001                  BCC     4$
8501
8502  027630  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8503
8504  027632  020304          4$:     CMP     R3,R4           ;RESULT = 77777?
8505  027634  001401                  BEQ     TST426          ;;BR IF YES
8506
8507  027636  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8508
8509                            ;;*******************************************************************
8510                            ;*TEST 426      SUB TEST - SM0,DM1 - N:C = 0110
8511                            ;;*******************************************************************
8512  027640                    TST426:
8513  027640  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8514  027642  012700  000426           MOV     #426,R0         ;;LOAD R0 WITH TEST NUMBER
8515  027646  013701  027674           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8516  027652  012702  063312           MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8517  027656  012704  177777           MOV     #-1,R4          ;RESULT S / B = 177777
8518  027662  012705  000001           MOV     #+1,R5          ;SRC OPR = +1
8519  027666  005012                   CLR     (R2)            ;[DEST] = 000000
8520  027670  000257                   CCC                     ;CLEAR FLAGS
8521  027672  000266                   266                     ;N:C = 0110
8522
8523  027674  160512          2$:     SUB     R5,(R2)         ;TEST THE SUB
8524
8525  027676  100003                  BPL     3$              ;N:C = 1001
8526  027700  001402                  BEQ     3$
8527  027702  102401                  BVS     3$
8528  027704  103401                  BCS     4$
```

```
8529
8530  027706  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8531
8532  027710  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8533  027712  001402                          BEQ     TST427          ;;BR IF YES
8534
8535  027714  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8536  027716  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8537
8538                                  ;;**********************************************************************
8539                                  ;*TEST 427       SUB TEST - SM0,DM1 - N:C = 1010
8540                                  ;;**********************************************************************
8541  027720                          TST427:
8542  027720  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8543  027722  012700  000427                  MOV     #427,R0         ;;LOAD R0 WITH TEST NUMBER
8544  027726  013701  027754                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8545  027732  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8546  027736  005004                          CLR     R4              ;RESULT S / B = 000000
8547  027740  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
8548  027744  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
8549  027750  000257                          CCC                     ;CLEAR FLAGS
8550  027752  000272                          272                     ;N:C = 1010
8551
8552  027754  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8553
8554  027756  100403                          BMI     3$              ;N:C = 0100
8555  027760  001002                          BNE     3$
8556  027762  102401                          BVS     3$
8557  027764  103001                          BCC     4$
8558
8559  027766  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8560
8561  027770  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8562  027772  001402                          BEQ     TST430          ;;BR IF YES
8563
8564  027774  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8565  027776  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8566
8567                                  ;;**********************************************************************
8568                                  ;*TEST 430       SUB TEST - SM0,DM1 - N:C = 0000
8569                                  ;;**********************************************************************
8570  030000                          TST430:
8571  030000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8572  030002  012700  000430                  MOV     #430,R0         ;;LOAD R0 WITH TEST NUMBER
8573  030006  013701  030034                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8574  030012  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8575  030016  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
8576  030022  012705  000001                  MOV     #+1,R5          ;SRC OPR = +1
8577  030026  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
8578  030032  000257                          CCC                     ;CLEAR FLAGS
8579
8580  030034  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8581
8582  030036  100403                          BMI     3$              ;N:C = 0010
8583  030040  001402                          BEQ     3$
8584  030042  102001                          BVC     3$
```

```
                        BCC     4$

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

                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
                        BEQ     TST431          ;;BR IF YES

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

                ;;**********************************************************************
                ;*TEST 431       SUB TEST - SM1,DM0 - N:C = 0110
                ;;**********************************************************************
030060          TST431:
030060  000004          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030062  012700  000431  MOV     #431,R0         ;;LOAD R0 WITH TEST NUMBER
030066  013701  030110  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030072  012704  177777  MOV     #-1,R4          ;RESULT S / B = 177777
030076  012705  064034  MOV     #DWTB+2,R5      ;SRC ADDR = DWTB+2
030102  005003          CLR     R3              ;[DEST] = 000000
030104  000257          CCC                     ;CLEAR FLAGS
030106  000266          266                     ;N:C = 0110

030110  161503  2$:     SUB     (R5),R3         ;TEST THE SUB

030112  100003          BPL     3$              ;N:C = 1001
030114  001402          BEQ     3$
030116  102401          BVS     3$
030120  103401          BCS     4$

030122  104002  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY

030124  020403  4$:     CMP     R4,R3           ;CORRECT RESULT ?
030126  001401          BEQ     TST432          ;;BR IF YES

030130  104002  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT

                ;;**********************************************************************
                ;*TEST 432       SUB TEST - SM1,DM0 - N:C = 1010
                ;;**********************************************************************
030132          TST432:
030132  000004          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030134  012700  000432  MOV     #432,R0         ;;LOAD R0 WITH TEST NUMBER
030140  013701  030160  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030144  005004          CLR     R4              ;RESULT S / B = 000000
030146  012705  063324  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
030152  011503          MOV     (R5),R3         ;[DEST] = 177777
030154  000257          CCC                     ;CLEAR FLAGS
030156  000272          272                     ;N:C = 1010

030160  161503  2$:     SUB     (R5),R3 ;TEST THE SUB

030162  100403          BMI     3$              ;N:C = 0100
030164  001002          BNE     3$
030166  102401          BVS     3$
030170  103001          BCC     4$
```

```
8641
8642  030172  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8643
8644  030174  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8645  030176  001401                          BEQ     TST433          ;;BR IF YES
8646
8647  030200  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8648
8649                                  ;;*****************************************************************
8650                                  ;*TEST 433       SUB TEST - SM1,DM0 - N:C = 0000
8651                                  ;;*****************************************************************
8652  030202                          TST433:
8653  030202  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8654  030204  012700  000433                  MOV     #433,R0         ;;LOAD R0 WITH TEST NUMBER
8655  030210  013701  030236                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8656  030214  012704  077777                  MOV     #77777,R4               ;RESULT S / B = 77777
8657  030220  012705  063316                  MOV     #MBUF1,R5               ;SRC ADDR =MBUF1
8658  030224  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
8659  030230  012715  000001                  MOV     #+1,(R5)        ;SRC OPR = +1
8660  030234  000257                          CCC                     ;CLEAR FLAGS
8661
8662  030236  161503                  2$:     SUB     (R5),R3         ;TEST THE SUB
8663
8664  030240  100403                          BMI     3$              ;N:C = 0010
8665  030242  001402                          BEQ     3$
8666  030244  102001                          BVC     3$
8667  030246  103001                          BCC     4$
8668
8669  030250  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8670
8671  030252  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
8672  030254  001401                          BEQ     TST434          ;;BR IF YES
8673
8674  030256  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8675
8676                                  ;;*****************************************************************
8677                                  ;*TEST 434       SUB  SM1,DM1 TEST - N:C = 0110
8678                                  ;;*****************************************************************
8679  030260                          TST434:
8680  030260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8681  030262  012700  000434                  MOV     #434,R0         ;;LOAD R0 WITH TEST NUMBER
8682  030266  013701  030320                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8683  030272  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8684  030276  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
8685  030302  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
8686  030306  012715  000001                  MOV     #+1,(R5)        ;[SOURCE] = 000001
8687  030312  005012                          CLR     (R2)            ;[DEST] = 000000
8688  030314  000257                          CCC                     ;CLEAR FLAGS
8689  030316  000266                          266                     ;N:C = 0110
8690
8691  030320  161512                  2$:     SUB     (R5),(R2)       ;TEST THE SUB
8692
8693  030322  100003                          BPL     3$              ;N:C = 1001 ?
8694  030324  001402                          BEQ     3$
8695  030326  102401                          BVS     3$
8696  030330  103401                          BCS     4$
```

```
8697
8698  030332  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8699
8700  030334  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8701  030336  001402                          BEQ     TST435          ;;BR IF YES
8702
8703  030340  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8704  030342  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8705
8706                                  ;;**********************************************************************
8707                                  ;*TEST 435       SUB  SM1,DM2 TEST - N:C = 0110
8708                                  ;;**********************************************************************
8709  030344                          TST435:
8710  030344  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8711  030346  012700  000435                  MOV     #435,R0         ;;LOAD R0 WITH TEST NUMBER
8712  030352  013701  030406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8713  030356  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8714  030362  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
8715  030366  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
8716  030372  012715  000001                  MOV     #+1,(R5)        ;[SOURCE] = 000001
8717  030376  005012                          CLR     (R2)            ;[DEST] = 000000
8718  030400  010203                          MOV     R2,R3           ;R3 GETS DEST ADDR
8719  030402  000257                          CCC                     ;CLEAR FLAGS
8720  030404  000266                          266                     ;N:C = 0110
8721
8722  030406  161523                  2$:     SUB     (R5),(R3)+      ;TEST THE SUB
8723
8724  030410  100003                          BPL     3$              ;N:C = 1001 ?
8725  030412  001402                          BEQ     3$
8726  030414  102401                          BVS     3$
8727  030416  103401                          BCS     4$
8728
8729  030420  104005                  3$:     ERROR   5               ;SUB FAILED TO ALTER CODES PROPERLY
8730
8731  030422  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8732  030424  001402                          BEQ     TST436          ;;BR IF YES
8733
8734  030426  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8735  030430  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8736
8737                                  ;;**********************************************************************
8738                                  ;*TEST 436       NEG DM2 TEST
8739                                  ;;**********************************************************************
8740  030432                          TST436:
8741  030432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8742  030434  012700  000436                  MOV     #436,R0         ;;LOAD R0 WITH TEST NUMBER
8743  030440  013701  030464                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8744  030444  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8745  030450  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
8746  030454  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
8747  030456  012712  052526                  MOV     #52526,(R2)     ;[DEST] = 52526
8748  030462  000257                          CCC                     ;SCOPE SYNC
8749
8750  030464  005425                  2$:     NEG     (R5)+           ;TEST THE NEG - MODE 2
8751
8752  030466  020412                          CMP     R4,(R2)         ;RESULT = 125252?
```

```
8753  030470  001402                  BEQ     4$              ;BR IF YES
8754
8755  030472  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8756  030474  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8757
8758  030476  022705  063314  4$:     CMP     #MBUF0+2,R5     ;DID REG. GET AUTO INCREMENTED?
8759  030502  001401                  BEQ     TST437          ;;BR IF YES
8760
8761  030504  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8762
8763                          ;;*****************************************************************
8764                          ;*TEST 437       NEG DM3 TEST
8765                          ;;*****************************************************************
8766  030506                  TST437:
8767  030506  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8768  030510  012700  000437          MOV     #437,R0         ;;LOAD R0 WITH TEST NUMBER
8769  030514  013701  030542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8770  030520  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8771  030524  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8772  030530  012705  063306          MOV     #ATA+10,R5      ;[ATA+10] = MBUF0
8773  030534  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8774  030540  000257                  CCC                     ;SCOPE SYNC
8775
8776  030542  005435          2$:     NEG     @(R5)+          ;TEST THE NEG - MODE 3
8777
8778  030544  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8779  030546  001402                  BEQ     4$              ;BR IF YES
8780
8781  030550  011203                  MOV     (R2),R3         ;GET WAS DATA
8782  030552  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8783
8784  030554  022705  063310  4$:     CMP     #ATA+12,R5      ;DID REG GET AUTO INCREMENTED?
8785  030560  001401                  BEQ     TST440          ;;BR IF YES
8786
8787  030562  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8788
8789                          ;;*****************************************************************
8790                          ;*TEST 440       NEG DM4 TEST
8791                          ;;*****************************************************************
8792  030564                  TST440:
8793  030564  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8794  030566  012700  000440          MOV     #440,R0         ;;LOAD R0 WITH TEST NUMBER
8795  030572  013701  030620          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8796  030576  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8797  030602  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8798  030606  012705  063314          MOV     #MBUF0+2,R5     ;[R5] = DEST ADDR + 2
8799  030612  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8800  030616  000257                  CCC                     ;SCOPE SYNC
8801
8802  030620  005445          2$:     NEG     -(R5)           ;TEST THE NEG - MODE 4
8803
8804  030622  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8805  030624  001402                  BEQ     4$              ;BR IF YES
8806
8807  030626  011203                  MOV     (R2),R3         ;GET WAS DATA
8808  030630  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
```

```
 8809
 8810  030632  020502          4$:     CMP     R5,R2           ;DID REG GET AUTO INCREMENTED?
 8811  030634  001401                  BEQ     TST441          ;;BR IF YES
 8812
 8813  030636  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
```

```
8814
8815                                    ;;*****************************************************************
8816                                    ;*TEST 441       NEG DM5 TEST
8817                                    ;;*****************************************************************
8818  030640                            TST441:
8819  030640  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8820  030642  012700  000441                   MOV     #441,R0         ;;LOAD R0 WITH TEST NUMBER
8821  030646  013701  030674                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8822  030652  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8823  030656  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8824  030662  012705  063310                   MOV     #ATA+12,R5      ;[R5] = (ADR OF MBUF0) +2
8825  030666  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8826  030672  000257                           CCC                     ;SCOPE SYNC
8827
8828  030674  005455                    2$:    NEG     @-(R5)          ;TEST THE NEG - MODE 5
8829
8830  030676  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8831  030700  001402                           BEQ     4$              ;BR IF YES
8832
8833  030702  011203                           MOV     (R2),R3         ;GET WAS DATA
8834  030704  104001                    3$:    ERROR   1               ;NEG DELIVERED WRONG RESULT
8835
8836  030706  022705  063306             4$:    CMP     #ATA+10,R5      ;DID NEG UPDATE REG
8837  030712  001401                           BEQ     TST442          ;;BR IF YES
8838
8839  030714  104005                    5$:    ERROR   5               ;NEG FAILED TO UPDATE REG
8840
8841                                    ;;*****************************************************************
8842                                    ;*TEST 442       NEG DM6 TEST
8843                                    ;;*****************************************************************
8844  030716                            TST442:
8845  030716  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8846  030720  012700  000442                   MOV     #442,R0         ;;LOAD R0 WITH TEST NUMBER
8847  030724  013701  030752                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8848  030730  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8849  030734  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8850  030740  012705  063310                   MOV     #MBUF0-2,R5     ;[R5] = BASE ADDR
8851  030744  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8852  030750  000257                           CCC                     ;SCOPE SYNC
8853
8854  030752  005465  000002             2$:    NEG     2(R5)           ;TEST THE NEG - MODE 6
8855
8856  030756  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8857  030760  001402                           BEQ     TST443          ;;BR IF YES
8858
8859  030762  011203                           MOV     (R2),R3         ;GET WAS DATA
8860  030764  104001                    3$:    ERROR   1               ;NEG DELIVERED WRONG RESULT
8861
8862                                    ;;*****************************************************************
8863                                    ;*TEST 443       NEG DM7 TEST
8864                                    ;;*****************************************************************
8865  030766                            TST443:
8866  030766  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8867  030770  012700  000443                   MOV     #443,R0         ;;LOAD R0 WITH TEST NUMBER
8868  030774  013701  031022                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8869  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
031020  000257                          CCC                     ;SCOPE SYNC

031022  005475  000010          2$:     NEG     @10(R5)         ;TEST THE NEG - MODE 7

031026  020412                          CMP     R4,(R2)         ;RESULT = 125252?
031030  001402                          BEQ     TST444          ;;BR IF YES

031032  011203                          MOV     (R2),R3         ;GET WAS DATA
031034  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT

                                ;;*******************************************************************
                                ;*TEST 444       MOV SM1,DM1 TEST - N:C = 0100
                                ;;*******************************************************************
031036                          TST444:
031036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
031040  012700  000444                  MOV     #444,R0         ;;LOAD R0 WITH TEST NUMBER
031044  013701  031074                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
031050  005004                          CLR     R4              ;RESULT S / B = 177777
031052  005104                          COM     R4
031054  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
031060  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
031064  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
031066  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
031070  000257                          CCC                     ;CLEAR FLAGS
031072  000264                          264                     ;N:C = 0100

031074  011513                  2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1

031076  100003                          BPL     3$              ;N:C = 1000 ?
031100  001402                          BEQ     3$
031102  102401                          BVS     3$
031104  103001                          BCC     4$

031106  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY

031110  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
031112  001403                          BEQ     TST445          ;;BR IF YES

031114  005003                          CLR     R3              ;GET THE WAS DATA
031116  051203                          BIS     (R2),R3
031120  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT

                                ;;*******************************************************************
                                ;*TEST 445       MOV SM2,DM1 TEST - N:C = 0100
                                ;;*******************************************************************
031122                          TST445:
031122  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
031124  012700  000445                  MOV     #445,R0         ;;LOAD R0 WITH TEST NUMBER
031130  013701  031160                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
031134  005004                          CLR     R4              ;RESULT S / B = 177777
031136  005104                          COM     R4
031140  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
031144  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
```

```
 8926  031150  010203                MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8927  031152  005012                CLR     (R2)            ;MAKE [DEST] = 000000
 8928  031154  000257                CCC                     ;CLEAR FLAGS
 8929  031156  000264                264                     ;N:C = 0100
 8930
 8931  031160  012513          2$:   MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
 8932
 8933  031162  100003                BPL     3$              ;N:C = 1000 ?
 8934  031164  001402                BEQ     3$
 8935  031166  102401                BVS     3$
 8936  031170  103001                BCC     4$
 8937
 8938  031172  104001          3$:   ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8939
 8940  031174  020412          4$:   CMP     R4,(R2)         ;RESULT CORRECT ??
 8941  031176  001403                BEQ     TST446          ;;BR IF YES
 8942
 8943  031200  005003                CLR     R3              ;GET THE WAS DATA
 8944  031202  051203                BIS     (R2),R3
 8945  031204  104001          5$:   ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8946
 8947                          ;;*****************************************************************
 8948                          ;*TEST 446      MOV SM1,DM1 TEST - N:C = 1011
 8949                          ;;*****************************************************************
 8950  031206                  TST446:
 8951  031206  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8952  031210  012700  000446        MOV     #446,R0         ;;LOAD R0 WITH TEST NUMBER
 8953  031214  013701  031244        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8954  031220  005004                CLR     R4              ;RESULT S / B = 000000
 8955  031222  012702  063312        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8956  031226  012705  063322        MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
 8957  031232  010203                MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8958  031234  005012                CLR     (R2)            ;MAKE [DEST] = 177777
 8959  031236  005112                COM     (R2)
 8960  031240  000257                CCC                     ;CLEAR FLAGS
 8961  031242  000273                273                     ;N:C = 1011
 8962
 8963  031244  011513          2$:   MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
 8964
 8965  031246  100403                BMI     3$              ;N:C = 0101 ?
 8966  031250  001002                BNE     3$
 8967  031252  102401                BVS     3$
 8968  031254  103401                BCS     4$
 8969
 8970  031256  104001          3$:   ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8971
 8972  031260  020412          4$:   CMP     R4,(R2)         ;RESULT CORRECT ??
 8973  031262  001403                BEQ     TST447          ;;BR IF YES
 8974
 8975  031264  005003                CLR     R3              ;GET THE WAS DATA
 8976  031266  051203                BIS     (R2),R3
 8977  031270  104001          5$:   ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8978
 8979                          ;;*****************************************************************
 8980                          ;*TEST 447      MOV SM2,DM1 TEST - N:C = 1011
 8981                          ;;*****************************************************************
```

```
 8982  031272
 8983  031272  000004
 8984  031274  012700  000447
 8985  031300  013701  031330
 8986  031304  005004
 8987  031306  012702  063312
 8988  031312  012705  063322
 8989  031316  010203
 8990  031320  005012
 8991  031322  005112
 8992  031324  000257
 8993  031326  000273
 8994
 8995  031330  012513
 8996
 8997  031332  100403
 8998  031334  001002
 8999  031336  102401
 9000  031340  103401
 9001
 9002  031342  104001
 9003
 9004  031344  020412
 9005  031346  001403
 9006
 9007  031350  005003
 9008  031352  051203
 9009  031354  104001
 9010
 9011
 9012
 9013
 9014  031356
 9015  031356  000004
 9016  031360  012700  000450
 9017  031364  013701  031414
 9018  031370  005004
 9019  031372  005104
 9020  031374  012702  063312
 9021  031400  012705  063324
 9022  031404  010203
 9023  031406  005012
 9024  031410  000257
 9025  031412  000264
 9026
 9027  031414  011523
 9028
 9029  031416  100003
 9030  031420  001402
 9031  031422  102401
 9032  031424  103001
 9033
 9034  031426  104001
 9035
 9036  031430  022703  063314
 9037  031434  001401

TST447:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #447,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        CLR     R4              ;RESULT S / B = 000000
        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
        MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
        MOV     R2,R3           ;BASE DEST ADDR = MBUF0
        CLR     (R2)            ;MAKE [DEST] = 177777
        COM     (R2)
        CCC                     ;CLEAR FLAGS
        273                     ;N:C = 1011
2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
        BMI     3$              ;N:C = 0101 ?
        BNE     3$
        BVS     3$
        BCS     4$
3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
        BEQ     TST450          ;;BR IF YES
        CLR     R3              ;GET THE WAS DATA
        BIS     (R2),R3
5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
;;**********************************************************************
;*TEST 450       MOV SM1,DM2 TEST - N:C = 0100
;;**********************************************************************
TST450:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #450,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        CLR     R4              ;RESULT S / B = 177777
        COM     R4
        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
        MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
        MOV     R2,R3           ;BASE DEST ADDR = MBUF0
        CLR     (R2)            ;MAKE [DEST] = 000000
        CCC                     ;CLEAR FLAGS
        264                     ;N:C = 0100
2$:     MOV     (R5),(R3)+      ;TEST THE MOV - SM1,DM2
        BPL     3$              ;N:C =  1000 ?
        BEQ     3$
        BVS     3$
        BCC     4$
3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
        BEQ     6$              ;BR IF YES
```

```
 9038
 9039  031436  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9040
 9041  031440  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9042  031442  001403                          BEQ     TST451          ;;BR IF YES
 9043
 9044  031444  005003                          CLR     R3              ;GET THE WAS DATA
 9045  031446  051203                          BIS     (R2),R3
 9046  031450  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9047
 9048                                  ;;*********************************************************************
 9049                                  ;*TEST 451      MOV SM2,DM2 TEST - N:C = 0100
 9050                                  ;;*********************************************************************
 9051  031452                          TST451:
 9052  031452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9053  031454  012700  000451                  MOV     #451,R0         ;;LOAD R0 WITH TEST NUMBER
 9054  031460  013701  031510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9055  031464  005004                          CLR     R4              ;RESULT S / B = 177777
 9056  031466  005104                          COM     R4
 9057  031470  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9058  031474  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
 9059  031500  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9060  031502  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9061  031504  000257                          CCC                     ;CLEAR FLAGS
 9062  031506  000264                          264                     ;N:C = 0100
 9063
 9064  031510  012523                  2$:     MOV     (R5)+,(R3)+     ;TEST THE MOV - SM2,DM2
 9065
 9066  031512  100003                          BPL     3$              ;N:C = 1000 ?
 9067  031514  001402                          BEQ     3$
 9068  031516  102401                          BVS     3$
 9069  031520  103001                          BCC     4$
 9070
 9071  031522  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9072
 9073  031524  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
 9074  031530  001401                          BEQ     6$              ;BR IF YES
 9075
 9076  031532  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9077
 9078  031534  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9079  031536  001403                          BEQ     TST452          ;;BR IF YES
 9080
 9081  031540  005003                          CLR     R3              ;GET THE WAS DATA
 9082  031542  051203                          BIS     (R2),R3
 9083  031544  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9084
 9085                                  ;;*********************************************************************
 9086                                  ;*TEST 452      MOV SM1,DM3 TEST - N:C = 0100
 9087                                  ;;*********************************************************************
 9088  031546                          TST452:
 9089  031546  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9090  031550  012700  000452                  MOV     #452,R0         ;;LOAD R0 WITH TEST NUMBER
 9091  031554  013701  031606                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9092  031560  005004                          CLR     R4              ;RESULT S / B = 177777
 9093  031562  005104                          COM     R4
```

```
 9094  031564  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9095  031570  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9096  031574  012703  063306          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9097  031600  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9098  031602  000257                  CCC                     ;CLEAR FLAGS
 9099  031604  000264                  264                     ;N:C = 0100
 9100
 9101  031606  011533          2$:     MOV     (R5),@(R3)+     ;TEST THE MOV - SM1,DM3
 9102
 9103  031610  100003                  BPL     3$              ;N:C = 1000 ?
 9104  031612  001402                  BEQ     3$
 9105  031614  102401                  BVS     3$
 9106  031616  103001                  BCC     4$
 9107
 9108  031620  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9109
 9110  031622  022703  063310  4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9111  031626  001401                  BEQ     6$              ;BR IF YES
 9112
 9113  031630  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9114
 9115  031632  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9116  031634  001403                  BEQ     TST453          ;;BR IF YES
 9117
 9118  031636  005003                  CLR     R3              ;GET THE WAS DATA
 9119  031640  051203                  BIS     (R2),R3
 9120  031642  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9121
 9122                          ;;*****************************************************************
 9123                          ;*TEST 453      MOV SM2,DM3 TEST - N:C = 0100
 9124                          ;;*****************************************************************
 9125  031644                  TST453:
 9126  031644  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9127  031646  012700  000453          MOV     #453,R0         ;;LOAD R0 WITH TEST NUMBER
 9128  031652  013701  031704          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9129  031656  005004                  CLR     R4              ;RESULT S / B = 177777
 9130  031660  005104                  COM     R4
 9131  031662  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9132  031666  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9133  031672  012703  063306          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9134  031676  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9135  031700  000257                  CCC                     ;CLEAR FLAGS
 9136  031702  000264                  264                     ;N:C = 0100
 9137
 9138  031704  012533          2$:     MOV     (R5)+,@(R3)+    ;TEST THE MOV - SM2,DM3
 9139
 9140  031706  100003                  BPL     3$              ;N:C = 1000 ?
 9141  031710  001402                  BEQ     3$
 9142  031712  102401                  BVS     3$
 9143  031714  103001                  BCC     4$
 9144
 9145  031716  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9146
 9147  031720  022703  063310  4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9148  031724  001401                  BEQ     6$              ;BR IF YES
 9149
```

```
 9150  031726  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9151
 9152  031730  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9153  031732  001403                        BEQ     TST454          ;;BR IF YES
 9154
 9155  031734  005003                        CLR     R3              ;GET THE WAS DATA
 9156  031736  051203                        BIS     (R2),R3
 9157  031740  104001                7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9158
 9159                                ;;*********************************************************************
 9160                                ;*TEST 454       MOV SM1,DM4 TEST - N:C = 0100
 9161                                ;;*********************************************************************
 9162  031742                        TST454:
 9163  031742  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9164  031744  012700  000454                MOV     #454,R0         ;;LOAD R0 WITH TEST NUMBER
 9165  031750  013701  032002                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9166  031754  005004                        CLR     R4              ;RESULT S / B = 177777
 9167  031756  005104                        COM     R4
 9168  031760  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9169  031764  012705  063324                MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9170  031770  012703  063314                MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
 9171  031774  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
 9172  031776  000257                        CCC                     ;CLEAR FLAGS
 9173  032000  000264                        264                     ;N:C = 0100
 9174
 9175  032002  011543                2$:     MOV     (R5),-(R3)      ;TEST THE MOV - SM1,DM4
 9176
 9177  032004  100003                        BPL     3$              ;N:C = 1000 ?
 9178  032006  001402                        BEQ     3$
 9179  032010  102401                        BVS     3$
 9180  032012  103001                        BCC     4$
 9181
 9182  032014  104001                3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9183
 9184  032016  020203                4$:     CMP     R2,R3           ;DID MOV DECREMENT DEST REG ?
 9185  032020  001401                        BEQ     6$              ;BR IF YES
 9186
 9187  032022  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9188
 9189  032024  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9190  032026  001403                        BEQ     TST455          ;;BR IF YES
 9191
 9192  032030  005003                        CLR     R3              ;GET THE WAS DATA
 9193  032032  051203                        BIS     (R2),R3
 9194  032034  104001                7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9195
 9196                                ;;*********************************************************************
 9197                                ;*TEST 455       MOV SM2,DM4 TEST - N:C = 0100
 9198                                ;;*********************************************************************
 9199  032036                        TST455:
 9200  032036  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9201  032040  012700  000455                MOV     #455,R0         ;;LOAD R0 WITH TEST NUMBER
 9202  032044  013701  032076                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9203  032050  005004                        CLR     R4              ;RESULT S / B = 177777
 9204  032052  005104                        COM     R4
 9205  032054  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
 9206  032060  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9207  032064  012703  063314          MOV     #MBUF0+2,R3             ;BASE DEST ADDR = MBUF0+2
 9208  032070  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9209  032072  000257                  CCC                     ;CLEAR FLAGS
 9210  032074  000264                  264                     ;N:C = 0100
 9211
 9212  032076  012543          2$:     MOV     (R5)+,-(R3)     ;TEST THE MOV - SM2,DM4
 9213
 9214  032100  100003                  BPL     3$              ;N:C = 1000 ?
 9215  032102  001402                  BEQ     3$
 9216  032104  102401                  BVS     3$
 9217  032106  103001                  BCC     4$
 9218
 9219  032110  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9220
 9221  032112  020203          4$:     CMP     R2,R3           ;DID MOV INCREMENT DEST REG ?
 9222  032114  001401                  BEQ     6$              ;BR IF YES
 9223
 9224  032116  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9225
 9226  032120  020412          6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9227  032122  001403                  BEQ     TST456          ;;BR IF YES
 9228
 9229  032124  005003                  CLR     R3              ;GET THE WAS DATA
 9230  032126  051203                  BIS     (R2),R3
 9231  032130  104001          7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9232
 9233                          ;;*****************************************************************
 9234                          ;*TEST 456       MOV SM1,DM5 TEST - N:C = 0100
 9235                          ;;*****************************************************************
 9236  032132                  TST456:
 9237  032132  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9238  032134  012700  000456          MOV     #456,R0         ;;LOAD R0 WITH TEST NUMBER
 9239  032140  013701  032204          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9240                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT9
 9241  032144  032737  001000  063234  BIT     #BIT9,@#BPTLOC  ;BREAKPOINT HALT SET ??
 9242  032152  001401                  BEQ     .+4             ;BR IF NOT
 9243  032154  000000                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 9244  032156  005004                  CLR     R4              ;RESULT S / B = 177777
 9245  032160  005104                  COM     R4
 9246  032162  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9247  032166  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9248  032172  012703  063310          MOV     #ATA+12,R3              ;BASE DEST ADDR = ATA+12
 9249  032176  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9250  032200  000257                  CCC                     ;CLEAR FLAGS
 9251  032202  000264                  264                     ;N:C = 0100
 9252
 9253  032204  011553          2$:     MOV     (R5),@-(R3)     ;TEST THE MOV - SM1,DM5
 9254
 9255  032206  100003                  BPL     3$              ;N:C = 0100 ?
 9256  032210  001402                  BEQ     3$
 9257  032212  102401                  BVS     3$
 9258  032214  103001                  BCC     4$
 9259
 9260  032216  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9261
```

```
 9262  032220  022703  063306      4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9263  032224  001401                      BEQ     6$              ;BR IF YES
 9264
 9265  032226  104005              5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9266
 9267  032230  020412              6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9268  032232  001403                      BEQ     TST457          ;;BR IF YES
 9269
 9270  032234  005003                      CLR     R3              ;GET THE WAS DATA
 9271  032236  051203                      BIS     (R2),R3
 9272  032240  104001              7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9273
 9274                              ;;*****************************************************************
 9275                              ;*TEST 457       MOV SM2,DM5 TEST - N:C = 0100
 9276                              ;;*****************************************************************
 9277  032242                      TST457:
 9278  032242  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9279  032244  012700  000457              MOV     #457,R0         ;;LOAD R0 WITH TEST NUMBER
 9280  032250  013701  032302              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9281  032254  005004                      CLR     R4              ;RESULT S / B = 177777
 9282  032256  005104                      COM     R4
 9283  032260  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9284  032264  012705  063324              MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9285  032270  012703  063310              MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
 9286  032274  005012                      CLR     (R2)            ;MAKE [DEST] = 00000
 9287  032276  000257                      CCC                     ;CLEAR FLAGS
 9288  032300  000264                      264                     ;N:C = 1000
 9289
 9290  032302  012553              2$:     MOV     (R5)+,@-(R3)    ;TEST THE MOV - SM2,DM5
 9291
 9292  032304  100003                      BPL     3$              ;N:C = 1000 ?
 9293  032306  001402                      BEQ     3$
 9294  032310  102401                      BVS     3$
 9295  032312  103001                      BCC     4$
 9296
 9297  032314  104001              3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9298
 9299  032316  022703  063306      4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9300  032322  001401                      BEQ     6$              ;BR IF YES
 9301
 9302  032324  104005              5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9303
 9304  032326  020412              6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9305  032330  001403                      BEQ     TST460          ;;BR IF YES
 9306
 9307  032332  005003                      CLR     R3              ;GET THE WAS DATA
 9308  032334  051203                      BIS     (R2),R3
 9309  032336  104001              7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9310
 9311                              ;;*****************************************************************
 9312                              ;*TEST 460       MOV SM1,DM6 TEST - N:C = 0100
 9313                              ;;*****************************************************************
 9314  032340                      TST460:
 9315  032340  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9316  032342  012700  000460              MOV     #460,R0         ;;LOAD R0 WITH TEST NUMBER
 9317  032346  013701  032400              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 9318  032352  005004                        CLR     R4              ;RESULT S / B = 177777
 9319  032354  005104                        COM     R4
 9320  032356  012702  063320                MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9321  032362  012705  063324                MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9322  032366  012703  063312                MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9323  032372  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
 9324  032374  000257                        CCC                     ;CLEAR FLAGS
 9325  032376  000264                        264                     ;N:C = 0100
 9326
 9327  032400  011563  000006        2$:     MOV     (R5),6(R3)      ;TEST THE MOV - SM1,DM6
 9328
 9329  032404  100003                        BPL     3$              ;N:C = 1000 ?
 9330  032406  001402                        BEQ     3$
 9331  032410  102401                        BVS     3$
 9332  032412  103001                        BCC     4$
 9333
 9334  032414  104001        3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9335
 9336  032416  020412        4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9337  032420  001403                BEQ     TST461          ;;BR IF YES
 9338
 9339  032422  005003                CLR     R3              ;GET THE WAS DATA
 9340  032424  051203                BIS     (R2),R3
 9341  032426  104001        5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9342
 9343                  ;;*****************************************************************
 9344                  ;*TEST 461       MOV SM2,DM6 TEST - N:C = 0100
 9345                  ;;*****************************************************************
 9346  032430        TST461:
 9347  032430  000004                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9348  032432  012700  000461        MOV     #461,R0         ;;LOAD R0 WITH TEST NUMBER
 9349  032436  013701  032470        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9350  032442  005004                CLR     R4              ;RESULT S / B = 177777
 9351  032444  005104                COM     R4
 9352  032446  012702  063320        MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9353  032452  012705  063324        MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9354  032456  012703  063312        MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9355  032462  005012                CLR     (R2)            ;MAKE [DEST] = 000000
 9356  032464  000257                CCC                     ;CLEAR FLAGS
 9357  032466  000264                264                     ;N:C = 0100
 9358
 9359  032470  012563  000006        2$:     MOV     (R5)+,6(R3)     ;TEST THE MOV - SM2,DM6
 9360
 9361  032474  100003                BPL     3$              ;N:C = 1000 ?
 9362  032476  001402                BEQ     3$
 9363  032500  102401                BVS     3$
 9364  032502  103001                BCC     4$
 9365
 9366  032504  104001        3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9367
 9368  032506  020412        4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9369  032510  001403                BEQ     TST462          ;;BR IF YES
 9370
 9371  032512  005003                CLR     R3              ;GET THE WAS DATA
 9372  032514  051203                BIS     (R2),R3
 9373  032516  104001        5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 462      MOV SM1,DM7 TEST - N:C = 0100
                                        ;;*****************************************************************
032520                                  TST462:
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     TST463          ;;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                                  TST463:
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$
```

```
9430  032664  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
9431
9432  032666  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
9433  032670  001403                          BEQ     TST464          ;;BR IF YES
9434
9435  032672  005003                          CLR     R3              ;GET THE WAS DATA
9436  032674  051203                          BIS     (R2),R3
9437  032676  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9438
9439                                  ;;*********************************************************************
9440                                  ;*TEST 464       MOV  SM0,DM1 TEST
9441                                  ;;*********************************************************************
9442  032700                          TST464:
9443  032700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9444  032702  012700  000464                  MOV     #464,R0         ;;LOAD R0 WITH TEST NUMBER
9445  032706  013701  032726                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9446  032712  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9447  032716  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9448  032720  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
9449  032722  005012                          CLR     (R2)            ;[DEST] = 000000
9450  032724  000257                          CCC                     ;SCOPE SYNC
9451
9452  032726  010015                  2$:     MOV     R0,(R5)         ;TEST THE MOV
9453
9454  032730  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9455  032732  001402                          BEQ     TST465          ;;BR IF YES
9456
9457  032734  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9458  032736  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9459
9460                                  ;;*********************************************************************
9461                                  ;*TEST 465       MOV  SM0,DM2 TEST
9462                                  ;;*********************************************************************
9463  032740                          TST465:
9464  032740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9465  032742  012700  000465                  MOV     #465,R0         ;;LOAD R0 WITH TEST NUMBER
9466  032746  013701  032766                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9467  032752  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9468  032756  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
9469  032760  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
9470  032762  005012                          CLR     (R2)            ;[DEST] = 000000
9471  032764  000257                          CCC                     ;SCOPE SYNC
9472
9473  032766  010025                  2$:     MOV     R0,(R5)+        ;TEST THE MOV
9474
9475  032770  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
9476  032772  001402                          BEQ     TST466          ;;BR IF YES
9477
9478  032774  011203                          MOV     (R2),R3         ;GET THE WAS DATA
9479  032776  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
9480
9481                                  ;;*********************************************************************
9482                                  ;*TEST 466       MOV  SM0,DM3 TEST
9483                                  ;;*********************************************************************
9484  033000                          TST466:
9485  033000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9486  033002  012700  000466          MOV     #466,R0         ;;LOAD R0 WITH TEST NUMBER
 9487  033006  013701  033030          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9488  033012  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9489  033016  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9490  033020  012705  063306          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9491  033024  005012                  CLR     (R2)            ;[DEST] = 000000
 9492  033026  000257                  CCC                     ;SCOPE SYNC
 9493
 9494  033030  010035          2$:     MOV     R0,@(R5)+       ;TEST THE MOV
 9495
 9496  033032  020412                  CMP     R4,(R2)         ;CORRECT RESULT
 9497  033034  001402                  BEQ     TST467          ;;BR IF YES
 9498
 9499  033036  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9500  033040  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9501
 9502                          ;;**********************************************************************
 9503                          ;*TEST 467       MOV  SM0,DM4 TEST
 9504                          ;;**********************************************************************
 9505  033042                  TST467:
 9506  033042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9507  033044  012700  000467          MOV     #467,R0         ;;LOAD R0 WITH TEST NUMBER
 9508  033050  013701  033072          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9509  033054  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9510  033060  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9511  033062  012705  063314          MOV     #MBUF0+2,R5     ;R5 CONTAINS BASE DEST ADDR
 9512  033066  005012                  CLR     (R2)            ;[DEST] = 000000
 9513  033070  000257                  CCC                     ;SCOPE SYNC
 9514
 9515  033072  010045          2$:     MOV     R0,-(R5)        ;TEST THE MOV
 9516
 9517  033074  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9518  033076  001402                  BEQ     TST470          ;;BR IF YES
 9519
 9520  033100  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9521  033102  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9522
 9523                          ;;**********************************************************************
 9524                          ;*TEST 470       MOV  SM0,DM5 TEST
 9525                          ;;**********************************************************************
 9526  033104                  TST470:
 9527  033104  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9528  033106  012700  000470          MOV     #470,R0         ;;LOAD R0 WITH TEST NUMBER
 9529  033112  013701  033134          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9530  033116  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9531  033122  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9532  033124  012705  063310          MOV     #ATA+12,R5      ;R5 CONTAINS BASE DEST ADDR
 9533  033130  005012                  CLR     (R2)            ;[DEST] = 000000
 9534  033132  000257                  CCC                     ;SCOPE SYNC
 9535
 9536  033134  010055          2$:     MOV     R0,@-(R5)       ;TEST THE MOV
 9537
 9538  033136  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9539  033140  001402                  BEQ     TST471          ;;BR IF YES
 9540
 9541  033142  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
 9542  033144  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9543
 9544
 9545                                  ;;*****************************************************************
 9546                                  ;*TEST 471       MOV  SM0,DM6 TEST
 9547                                  ;;*****************************************************************
 9548  033146                          TST471:
 9549  033146  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9550  033150  012700  000471                  MOV     #471,R0         ;;LOAD R0 WITH TEST NUMBER
 9551  033154  013701  033176                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9552  033160  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 9553  033164  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9554  033166  012705  063312                  MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
 9555  033172  005012                          CLR     (R2)            ;[DEST] = 000000
 9556  033174  000257                          CCC                     ;SCOPE SYNC
 9557
 9558  033176  010065  000004          2$:     MOV     R0,4(R5)        ;TEST THE MOV
 9559
 9560  033202  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9561  033204  001402                          BEQ     TST472          ;;BR IF YES
 9562
 9563  033206  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9564  033210  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9565
 9566                                  ;;*****************************************************************
 9567                                  ;*TEST 472       MOVB TEST - SM0,DM0 - EXTEND 1'S
 9568                                  ;;*****************************************************************
 9569  033212                          TST472:
 9570  033212  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9571  033214  012700  000472                  MOV     #472,R0         ;;LOAD R0 WITH TEST NUMBER
 9572  033220  013701  033242                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9573  033224  012704  177652                  MOV     #177652,R4      ;RESULT S / B = 177652
 9574  033230  012705  000252                  MOV     #252,R5         ;SOURCE OP = 252
 9575  033234  005003                          CLR     R3              ;[DEST] = 000000
 9576  033236  000257                          CCC                     ;CLEAR FLAGS
 9577  033240  000266                          266                     ;N:C = 0110
 9578
 9579  033242  110503                  2$:     MOVB    R5,R3           ;TEST THE MOVB
 9580
 9581  033244  100003                          BPL     3$              ;N:C = 1000 ?
 9582  033246  001402                          BEQ     3$
 9583  033250  102401                          BVS     3$
 9584  033252  103001                          BCC     4$
 9585
 9586  033254  104002                  3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9587
 9588  033256  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9589  033260  001401                          BEQ     TST473          ;;BR IF YES
 9590
 9591  033262  104002                  5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9592
 9593                                  ;;*****************************************************************
 9594                                  ;*TEST 473       MOVB TEST - SM0,DM0 - EXTEND 0'S
 9595                                  ;;*****************************************************************
 9596  033264                          TST473:
 9597  033264  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9598  033266  012700  000473          MOV     #473,R0         ;;LOAD R0 WITH TEST NUMBER
 9599  033272  013701  033314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9600  033276  005004                  CLR     R4              ;RESULT S / B = 000000
 9601  033300  012705  177400          MOV     #177400,R5      ;SOURCE OP = 177400
 9602  033304  005003                  CLR     R3              ;[DEST] = 177777
 9603  033306  005103                  COM     R3
 9604  033310  000257                  CCC                     ;CLEAR FLAGS
 9605  033312  000271                  271                     ;N:C = 1001
 9606
 9607  033314  110503          2$:     MOVB    R5,R3           ;TEST THE MOVB
 9608
 9609  033316  100403                  BMI     3$              ;N:C = 0101 ?
 9610  033320  001002                  BNE     3$
 9611  033322  102401                  BVS     3$
 9612  033324  103401                  BCS     4$
 9613
 9614  033326  104002          3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9615
 9616  033330  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9617  033332  001401                  BEQ     TST474          ;;BR IF YES
 9618
 9619  033334  104002          5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9620
 9621                          ;;*********************************************************************
 9622                          ;*TEST 474       MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
 9623                          ;;*********************************************************************
 9624  033336                  TST474:
 9625  033336  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9626  033340  012700  000474          MOV     #474,R0         ;;LOAD R0 WITH TEST NUMBER
 9627  033344  013701  033364          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9628  033350  005004                  CLR     R4              ;RESULT S / B = 000000
 9629  033352  012705  064630          MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
 9630  033356  005003                  CLR     R3              ;[DEST] = 177777
 9631  033360  005103                  COM     R3
 9632  033362  000257                  CCC                     ;SCOPE SYNC
 9633
 9634  033364  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
 9635
 9636  033366  020403                  CMP     R4,R3           ;RESULT CORRECT ?
 9637  033370  001401                  BEQ     TST475          ;;BR IF YES
 9638
 9639  033372  104002          3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9640
 9641                          ;;*********************************************************************
 9642                          ;*TEST 475       MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
 9643                          ;;*********************************************************************
 9644  033374                  TST475:
 9645  033374  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9646  033376  012700  000475          MOV     #475,R0         ;;LOAD R0 WITH TEST NUMBER
 9647  033402  013701  033424          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9648  033406  012704  000125          MOV     #125,R4         ;RESULT S / B = 125
 9649  033412  012705  064633          MOV     #DBTA+3,R5      ;SOURCE ADDR = DBTA+3
 9650  033416  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
 9651  033422  000257                  CCC                     ;SCOPE SYNC
 9652
 9653  033424  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
```

```
9654
9655  033426  020403                    CMP     R4,R3           ;RESULT CORRECT ?
9656  033430  001401                    BEQ     TST476          ;;BR IF YES
9657
9658  033432  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9659
9660                                    ;;*********************************************************************
9661                                    ;*TEST 476       MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
9662                                    ;;*********************************************************************
9663  033434                            TST476:
9664  033434  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9665  033436  012700  000476                    MOV     #476,R0         ;;LOAD R0 WITH TEST NUMBER
9666  033442  013701  033462                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9667  033446  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
9668  033452  012705  064631                    MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
9669  033456  005003                            CLR     R3              ;[DEST] = 000000
9670  033460  000257                            CCC                     ;SCOPE SYNC
9671
9672  033462  112503            2$:     MOVB    (R5)+,R3        ;TEST THE MOVB
9673
9674  033464  020403                    CMP     R4,R3           ;RESULT CORRECT ?
9675  033466  001401                    BEQ     4$              ;BR IF YES
9676
9677  033470  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9678
9679  033472  022705  064632    4$:     CMP     #DBTA+2,R5      ;DID MOVB INCREMENT SRC REG ?
9680  033476  001401                    BEQ     TST477          ;;BR IF YES
9681
9682  033500  104005            5$:     ERROR   5               ;MOVB FAILED TO UPDATE SRC REG
9683
9684                                    ;;*********************************************************************
9685                                    ;*TEST 477       MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
9686                                    ;;*********************************************************************
9687  033502                            TST477:
9688  033502  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9689  033504  012700  000477                    MOV     #477,R0         ;;LOAD R0 WITH TEST NUMBER
9690  033510  013701  033530                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9691  033514  005004                            CLR     R4              ;RESULT S / B = 000000
9692  033516  012705  064630                    MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
9693  033522  012703  177400                    MOV     #177400,R3      ;[DEST] = 177400
9694  033526  000257                            CCC                     ;SCOPE SYNC
9695
9696  033530  112503            2$:     MOVB    (R5)+,R3                ;TEST THE MOVB
9697
9698  033532  020403                    CMP     R4,R3           ;RESULT CORRECT ?
9699  033534  001401                    BEQ     4$              ;BR IF YES
9700
9701  033536  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
9702
9703  033540  022705  064631    4$:     CMP     #DBTA+1,R5      ;DID MOVB INCREMENT SRC REG ?
9704  033544  001401                    BEQ     TST500          ;;BR IF YES
9705
9706  033546  104005            5$:     ERROR   5               ;MOVB FAILED TO UPDATE SOURCE REG
9707
9708                                    ;;*********************************************************************
9709                                    ;*TEST 500       MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
```

```
                                        ;;*****************************************************************
033550                                  TST500:
033550  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
033552  012700  000500                          MOV     #500,R0         ;;LOAD R0 WITH TEST NUMBER
033556  013701  033602                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
033562  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
033566  012704  000377                          MOV     #377,R4         ;RESULT S / B = 377
033572  012705  064631                          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
033576  005012                                  CLR     (R2)            ;[DEST] = 000000
033600  000257                                  CCC                     ;CLEAR FLAGS - SCOPE SYNC

033602  111512                          2$:     MOVB    (R5),(R2)       ;TEST THE MOVB

033604  020412                                  CMP     R4,(R2)         ;CORRECT RESULT ?
033606  001402                                  BEQ     TST501          ;;BR IF YES

033610  011203                                  MOV     (R2),R3         ;GET THE WAS DATA
033612  104001                          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT

                                        ;;*****************************************************************
                                        ;*TEST 501      MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
                                        ;;*****************************************************************
033614                                  TST501:
033614  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
033616  012700  000501                          MOV     #501,R0         ;;LOAD R0 WITH TEST NUMBER
033622  013701  033650                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
033626  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
033632  012704  000377                          MOV     #377,R4         ;RESULT S / B = 377
033636  012705  064631                          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
033642  005012                                  CLR     (R2)            ;[DEST] = 000000
033644  010203                                  MOV     R2,R3           ;[R3] = DEST ADDR
033646  000257                                  CCC                     ;CLEAR FLAGS - SCOPE SYNC

033650  111523                          2$:     MOVB    (R5),(R3)+      ;TEST THE MOVB

033652  020412                                  CMP     R4,(R2)         ;CORRECT RESULT ?
033654  001402                                  BEQ     4$              ;BR IF YES

033656  011203                                  MOV     (R2),R3         ;GET THE WAS DATA
033660  104001                          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT

033662  022703  063313                  4$:     CMP     #MBUF0+1,R3     ;DID MOVB INCREMENT THE DEST REG ?
033666  001401                                  BEQ     TST502          ;;BR IF YES

033670  104005                          5$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG

                                        ;;*****************************************************************
                                        ;*TEST 502      MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
                                        ;;*****************************************************************
033672                                  TST502:
033672  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
033674  012700  000502                          MOV     #502,R0         ;;LOAD R0 WITH TEST NUMBER
033700  013701  033730                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
033704  012702  063312                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
033710  012704  000377                          MOV     #377,R4         ;RESULT S / B = 377
033714  012705  064631                          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
```

```
 9766  033720  005012                    CLR     (R2)            ;[DEST] = 000000
 9767  033722  012703  063306            MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA +10
 9768  033726  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9769
 9770  033730  111533            2$:     MOVB    (R5),@(R3)+     ;TEST THE MOVB
 9771
 9772  033732  022703  063310            CMP     #ATA+12,R3      ;DID DEST REG GET INCREMENTED ?
 9773  033736  001401                    BEQ     4$              ;BR IF YES
 9774
 9775  033740  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9776
 9777  033742  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9778  033744  001402                    BEQ     TST503          ;;BR IF YES
 9779
 9780  033746  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9781  033750  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9782
 9783                            ;;*****************************************************************
 9784                            ;*TEST 503      MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
 9785                            ;;*****************************************************************
 9786  033752                    TST503:
 9787  033752  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9788  033754  012700  000503            MOV     #503,R0         ;;LOAD R0 WITH TEST NUMBER
 9789  033760  013701  034010            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9790  033764  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9791  033770  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9792  033774  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9793  034000  005012                    CLR     (R2)            ;[DEST] = 000000
 9794  034002  012703  063313            MOV     #MBUF0+1,R3     ;INITIAL DEST ADDR = MBUF0+1
 9795  034006  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9796
 9797  034010  111543            2$:     MOVB    (R5),-(R3)      ;TEST THE MOVB
 9798
 9799  034012  020302                    CMP     R3,R2           ;DID MOVB DECREMENT DEST REG ?
 9800  034014  001401                    BEQ     4$              ;BR IF YES
 9801
 9802  034016  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9803
 9804  034020  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9805  034022  001402                    BEQ     TST504          ;;BR IF YES
 9806
 9807  034024  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9808  034026  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9809
 9810                            ;;*****************************************************************
 9811                            ;*TEST 504      MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
 9812                            ;;*****************************************************************
 9813  034030                    TST504:
 9814  034030  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9815  034032  012700  000504            MOV     #504,R0         ;;LOAD R0 WITH TEST NUMBER
 9816  034036  013701  034066            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9817  034042  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9818  034046  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9819  034052  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9820  034056  005012                    CLR     (R2)            ;[DEST] = 000000
 9821  034060  012703  063310            MOV     #ATA+12,R3      ;INITIAL DEST ADDR = ATA +12
```

```
 9822  034064  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9823
 9824  034066  111553            2$:     MOVB    (R5),@-(R3)     ;TEST THE MOVB
 9825
 9826  034070  022703  063306            CMP     #ATA+10,R3      ;DID MOVB DECREMENT DEST REG ?
 9827  034074  001401                    BEQ     4$              ;BR IF YES
 9828
 9829  034076  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9830
 9831  034100  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9832  034102  001402                    BEQ     TST505          ;;BR IF YES
 9833
 9834  034104  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9835  034106  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9836
 9837                            ;;*****************************************************************
 9838                            ;*TEST 505      MOVB TEST - SM1,DM6 - SRC ADR ODD / DST ADR EVEN
 9839                            ;;*****************************************************************
 9840  034110                    TST505:
 9841  034110  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9842  034112  012700  000505            MOV     #505,R0         ;;LOAD R0 WITH TEST NUMBER
 9843  034116  013701  034146            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9844  034122  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9845  034126  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9846  034132  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9847  034136  005012                    CLR     (R2)            ;[DEST] = 000000
 9848  034140  012703  063320            MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
 9849  034144  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9850
 9851  034146  111563  177772    2$:     MOVB    (R5),-6(R3)     ;TEST THE MOVB
 9852
 9853  034152  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9854  034154  001402                    BEQ     TST506          ;;BR IF YES
 9855
 9856  034156  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9857  034160  104001            3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9858
 9859                            ;;*****************************************************************
 9860                            ;*TEST 506      MOVB TEST - SM1,DM7 - SRC ADR ODD / DST ADR EVEN
 9861                            ;;*****************************************************************
 9862  034162                    TST506:
 9863  034162  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9864  034164  012700  000506            MOV     #506,R0         ;;LOAD R0 WITH TEST NUMBER
 9865  034170  013701  034220            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9866  034174  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9867  034200  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9868  034204  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9869  034210  005012                    CLR     (R2)            ;[DEST] = 000000
 9870  034212  012703  063276            MOV     #ATA,R3         ;BASE DEST ADDR = ATA
 9871  034216  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9872
 9873  034220  111573  000010    2$:     MOVB    (R5),@10(R3)    ;TEST THE MOVB
 9874
 9875  034224  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9876  034226  001402                    BEQ     TST507          ;;BR IF YES
 9877
```

```
 9878  034230  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9879  034232  104001                3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9880
 9881                                        ;;*****************************************************************
 9882                                        ;*TEST 507      MOVB  SM0,DM1 TEST
 9883                                        ;;*****************************************************************
 9884  034234                        TST507:
 9885  034234  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9886  034236  012700  000507                MOV     #507,R0         ;;LOAD R0 WITH TEST NUMBER
 9887  034242  013701  034270                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9888  034246  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9889  034252  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9890  034256  012703  177777                MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9891  034262  010205                        MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9892  034264  005012                        CLR     (R2)            ;[DEST] = 000000
 9893  034266  000257                        CCC                     ;SCOPE SYNC
 9894
 9895  034270  110315                2$:     MOVB    R3,(R5)         ;TEST THE MOVB
 9896
 9897  034272  020412                        CMP     R4,(R2)         ;RESULT CORRECT ?
 9898  034274  001402                        BEQ     TST510          ;;BR IF YES
 9899
 9900  034276  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9901  034300  104001                3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9902
 9903                                        ;;*****************************************************************
 9904                                        ;*TEST 510      MOVB  SM0,DM2 TEST
 9905                                        ;;*****************************************************************
 9906  034302                        TST510:
 9907  034302  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9908  034304  012700  000510                MOV     #510,R0         ;;LOAD R0 WITH TEST NUMBER
 9909  034310  013701  034336                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9910  034314  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9911  034320  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
 9912  034324  012703  177777                MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9913  034330  010205                        MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9914  034332  005012                        CLR     (R2)            ;[DEST] = 000000
 9915  034334  000257                        CCC                     ;SCOPE SYNC
 9916
 9917  034336  110325                2$:     MOVB    R3,(R5)+        ;TEST THE MOVB
 9918
 9919  034340  020412                        CMP     R4,(R2)         ;RESULT CORRECT ?
 9920  034342  001402                        BEQ     TST511          ;;BR IF YES
 9921
 9922  034344  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9923  034346  104001                3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9924
 9925                                        ;;*****************************************************************
 9926                                        ;*TEST 511      MOVB  SM0,DM3 TEST
 9927                                        ;;*****************************************************************
 9928  034350                        TST511:
 9929  034350  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9930  034352  012700  000511                MOV     #511,R0         ;;LOAD R0 WITH TEST NUMBER
 9931  034356  013701  034406                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9932  034362  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9933  034366  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
```

```
 9934  034372  012703  177777          MOV     #-1,R3          ;SOURCE OP IN R3
 9935  034376  012705  063306          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9936  034402  005012                  CLR     (R2)            ;[DEST] = 000000
 9937  034404  000257                  CCC                     ;SCOPE SYNC
 9938
 9939  034406  110335          2$:     MOVB    R3,@(R5)+       ;TEST THE MOVB
 9940
 9941  034410  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9942  034412  001402                  BEQ     TST512          ;;BR IF YES
 9943
 9944  034414  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9945  034416  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9946
 9947                          ;;**************************************************************
 9948                          ;*TEST 512       MOVB  SM0,DM4 TEST
 9949                          ;;**************************************************************
 9950  034420                  TST512:
 9951  034420  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9952  034422  012700  000512          MOV     #512,R0         ;;LOAD R0 WITH TEST NUMBER
 9953  034426  013701  034456          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9954  034432  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9955  034436  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
 9956  034442  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9957  034446  012705  063314          MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9958  034452  005012                  CLR     (R2)            ;[DEST] = 000000
 9959  034454  000257                  CCC                     ;SCOPE SYNC
 9960
 9961  034456  110345          2$:     MOVB    R3,-(R5)        ;TEST THE MOVB
 9962
 9963  034460  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9964  034462  001402                  BEQ     TST513          ;;BR IF YES
 9965
 9966  034464  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9967  034466  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9968
 9969                          ;;**************************************************************
 9970                          ;*TEST 513       MOVB  SM0,DM6 TEST
 9971                          ;;**************************************************************
 9972  034470                  TST513:
 9973  034470  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9974  034472  012700  000513          MOV     #513,R0         ;;LOAD R0 WITH TEST NUMBER
 9975  034476  013701  034526          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9976  034502  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9977  034506  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9978  034512  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9979  034516  012705  063314          MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9980  034522  005012                  CLR     (R2)            ;[DEST] = 000000
 9981  034524  000257                  CCC                     ;SCOPE SYNC
 9982
 9983  034526  110365  177776  2$:     MOVB    R3,-2(R5)       ;TEST THE MOVB
 9984
 9985  034532  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9986  034534  001402                  BEQ     TST514          ;;BR IF YES
 9987
 9988  034536  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9989  034540  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 514      BIS TEST - SM0,DM0 - N:C = 0111
                                        ;;*****************************************************************
034542                                  TST514:
034542  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
034544  012700  000514                          MOV     #514,R0         ;;LOAD R0 WITH TEST NUMBER
034550  013701  034574                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
034554  012704  177777                          MOV     #-1,R4          ;RESULT S / B = 177777
034560  012705  125252                          MOV     #125252,R5      ;SRC OPR = 125252
034564  012703  052525                          MOV     #52525,R3       ;[DEST] = 52525
034570  000257                                  CCC                     ;CLEAR FLAGS
034572  000267                                  267                     ;N:C = 0111

034574  050503                          2$:     BIS     R5,R3           ;TEST THE BIS

034576  100003                                  BPL     3$              ;N:C = 1001 ?
034600  001402                                  BEQ     3$
034602  102401                                  BVS     3$
034604  103401                                  BCS     4$

034606  104002                          3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY

034610  020403                          4$:     CMP     R4,R3           ;CORRECT RESULT ?
034612  001401                                  BEQ     TST515          ;;BR IF YES

034614  104002                          5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT

                                        ;;*****************************************************************
                                        ;*TEST 515      BIS TEST - SM0,DM0 - N:C = 1000
                                        ;;*****************************************************************
034616                                  TST515:
034616  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
034620  012700  000515                          MOV     #515,R0         ;;LOAD R0 WITH TEST NUMBER
034624  013701  034654                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        .SBTTL USER CONTROLLED BREAKPOINT -- BIT10
034630  032737  002000  063234                  BIT     #BIT10,@#BPTLOC ;BREAKPOINT HALT SET ??
034636  001401                                  BEQ     .+4             ;BR IF NOT
034640  000000                                  HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
034642  005004                                  CLR     R4              ;RESULT S / B = 000000
034644  005005                                  CLR     R5              ;SRC OPR = 000000
034646  005003                                  CLR     R3              ;[DEST] = 000000
034650  000257                                  CCC                     ;CLEAR FLAGS
034652  000270                                  SEN                     ;N:C = 1000

034654  050503                          2$:     BIS     R5,R3           ;TEST THE BIS

034656  100403                                  BMI     3$              ;N:C = 0100
034660  001002                                  BNE     3$
034662  102401                                  BVS     3$
034664  103001                                  BCC     4$

034666  104002                          3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY

034670  020403                          4$:     CMP     R4,R3           ;CORRECT RESULT ?
034672  001401                                  BEQ     TST516          ;;BR IF YES
```

```
10046
10047  034674  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10048
10049                                  ;;*****************************************************************
10050                                  ;*TEST 516      BIC TEST - SM0,DM0 - N:C = 0111
10051                                  ;;*****************************************************************
10052  034676                          TST516:
10053  034676  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10054  034700  012700  000516                 MOV     #516,R0         ;;LOAD R0 WITH TEST NUMBER
10055  034704  013701  034730                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10056  034710  012704  100000                 MOV     #100000,R4      ;RESULT S / B = 100000
10057  034714  012705  077777                 MOV     #77777,R5       ;SRC OPR = 77777
10058  034720  012703  177777                 MOV     #-1,R3          ;[DEST] = 177777
10059  034724  000257                         CCC                     ;CLEAR FLAGS
10060  034726  000267                         267                     ;N:C = 0111
10061
10062  034730  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10063
10064  034732  100003                         BPL     3$              ;N:C = 1001 ?
10065  034734  001402                         BEQ     3$
10066  034736  102401                         BVS     3$
10067  034740  103401                         BCS     4$
10068
10069  034742  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10070
10071  034744  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10072  034746  001401                         BEQ     TST517          ;;BR IF YES
10073
10074  034750  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10075
10076                                  ;;*****************************************************************
10077                                  ;*TEST 517      BIC TEST - SM0,DM0 - N:C = 1000
10078                                  ;;*****************************************************************
10079  034752                          TST517:
10080  034752  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10081  034754  012700  000517                 MOV     #517,R0         ;;LOAD R0 WITH TEST NUMBER
10082  034760  013701  034776                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10083  034764  005004                         CLR     R4              ;RESULT S / B = 000000
10084  034766  005005                         CLR     R5              ;SRC OPR = 000000
10085  034770  005003                         CLR     R3              ;[DEST] = 000000
10086  034772  000257                         CCC                     ;CLEAR FLAGS
10087  034774  000270                         SEN                     ;N:C = 1000
10088
10089  034776  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10090
10091  035000  100403                         BMI     3$              ;N:C = 0100
10092  035002  001002                         BNE     3$
10093  035004  102401                         BVS     3$
10094  035006  103001                         BCC     4$
```

```
10095
10096  035010  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10097
10098  035012  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10099  035014  001401                          BEQ     TST520          ;;BR IF YES
10100
10101  035016  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10102
10103                                  ;;**********************************************************************
10104                                  ;*TEST 520       BIT TEST - SM0,DM0 - N:C = 0111
10105                                  ;;**********************************************************************
10106  035020                          TST520:
10107  035020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10108  035022  012700  000520                  MOV     #520,R0         ;;LOAD R0 WITH TEST NUMBER
10109  035026  013701  035052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10110  035032  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10111  035036  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10112  035042  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
10113  035046  000257                          CCC                     ;CLEAR FLAGS
10114  035050  000267                          267                     ;N:C = 0111
10115
10116  035052  030503                  2$:     BIT     R5,R3           ;TEST THE BIT
10117
10118  035054  100003                          BPL     3$              ;N:C = 1001
10119  035056  001402                          BEQ     3$
10120  035060  102401                          BVS     3$
10121  035062  103401                          BCS     4$
10122
10123  035064  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10124
10125  035066  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10126  035070  001402                          BEQ     TST521          ;;BR IF YES
10127
10128  035072  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10129  035074  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10130
10131                                  ;;**********************************************************************
10132                                  ;*TEST 521       BIT TEST - SM0,DM0 - N:C = 1000
10133                                  ;;**********************************************************************
10134  035076                          TST521:
10135  035076  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10136  035100  012700  000521                  MOV     #521,R0         ;;LOAD R0 WITH TEST NUMBER
10137  035104  013701  035126                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10138  035110  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
10139  035114  012705  052525                  MOV     #52525,R5       ;SRC OPR = 52525
10140  035120  010403                          MOV     R4,R3           ;[DEST] = 125252
10141  035122  000257                          CCC                     ;CLEAR FLAGS
10142  035124  000270                          SEN                     ;N:C = 1000
10143
10144  035126  030503                  2$:     BIT     R5,R3   ;TEST THE BIT
10145
10146  035130  100403                          BMI     3$              ;N:C = 0100
10147  035132  001002                          BNE     3$
10148  035134  102401                          BVS     3$
10149  035136  103001                          BCC     4$
10150
```

```
10151  035140  104002              3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10152
10153  035142  020403              4$:     CMP     R4,R3           ;CORRECT RESULT ?
10154  035144  001401                      BEQ     TST522          ;;BR IF YES
10155
10156  035146  104002              5$:     ERROR   2               ;BIT DELIVERED A RESULT
10157
10158                              ;;**********************************************************************
10159                              ;*TEST 522       CMP TEST - SM0,DM0 - N:C = 0110
10160                              ;;**********************************************************************
10161  035150                      TST522:
10162  035150  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10163  035152  012700  000522              MOV     #522,R0         ;;LOAD R0 WITH TEST NUMBER
10164  035156  013701  035200              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10165  035162  012704  000001              MOV     #+1,R4          ;RESULT S / B = +1
10166  035166  005005                      CLR     R5              ;SRC OPR = 000000
10167  035170  012703  000001              MOV     #+1,R3          ;[DEST] = +1
10168  035174  000257                      CCC                     ;CLEAR FLAGS
10169  035176  000266                      266                     ;N:C = 0110
10170
10171  035200  020503              2$:     CMP     R5,R3           ;TEST THE CMP
10172
10173  035202  100003                      BPL     3$              ;N:C = 1001
10174  035204  001402                      BEQ     3$
10175  035206  102401                      BVS     3$
10176  035210  103401                      BCS     4$
10177
10178  035212  104002              3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10179
10180  035214  020403              4$:     CMP     R4,R3           ;CORRECT RESULT ?
10181  035216  001401                      BEQ     TST523          ;;BR IF YES
10182
10183  035220  104002              5$:     ERROR   2               ;CMP DELIVERED A RESULT
10184
10185                              ;;**********************************************************************
10186                              ;*TEST 523       CMP TEST - SM0,DM0 - N:C = 1010
10187                              ;;**********************************************************************
10188  035222                      TST523:
10189  035222  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10190  035224  012700  000523              MOV     #523,R0         ;;LOAD R0 WITH TEST NUMBER
10191  035230  013701  035252              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10192  035234  012704  177777              MOV     #-1,R4          ;RESULT S / B = 177777
10193  035240  012705  177777              MOV     #-1,R5          ;SRC OPR = 177777
10194  035244  010403                      MOV     R4,R3           ;[DEST] = 177777
10195  035246  000257                      CCC                     ;CLEAR FLAGS
10196  035250  000272                      272                     ;N:C = 1010
10197
10198  035252  020503              2$:     CMP     R5,R3           ;TEST THE CMP
10199
10200  035254  100403                      BMI     3$              ;N:C = 0100
10201  035256  001002                      BNE     3$
10202  035260  102401                      BVS     3$
10203  035262  103001                      BCC     4$
10204
10205  035264  104002              3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10206
```

```
10207  035266  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10208  035270  001401                         BEQ     TST524          ;;BR IF YES
10209
10210  035272  104002                 5$:     ERROR   2               ;CMP DELIVERED A RESULT
10211
10212                                 ;;*********************************************************************
10213                                 ;*TEST 524       CMP TEST - SM0,DM0 - N:C = 0000
10214                                 ;;*********************************************************************
10215  035274                         TST524:
10216  035274  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10217  035276  012700  000524                 MOV     #524,R0         ;;LOAD R0 WITH TEST NUMBER
10218  035302  013701  035324                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10219  035306  012704  000001                 MOV     #+1,R4          ;RESULT S / B = +1
10220  035312  012705  100000                 MOV     #100000,R5                      ;SRC OPR = 100000
10221  035316  012703  000001                 MOV     #+1,R3          ;[DEST] = +1
10222  035322  000257                         CCC                     ;CLEAR FLAGS
10223
10224  035324  020503                 2$:     CMP     R5,R3           ;TEST THE CMP
10225
10226  035326  100403                         BMI     3$              ;N:C = 0010
10227  035330  001402                         BEQ     3$
10228  035332  102001                         BVC     3$
10229  035334  103001                         BCC     4$
10230
10231  035336  104002                 3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10232
10233  035340  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10234  035342  001401                         BEQ     TST525          ;;BR IF YES
10235
10236  035344  104002                 5$:     ERROR   2               ;CMP DELIVERED A RESULT
10237
10238                                 ;;*********************************************************************
10239                                 ;*TEST 525       BIS TEST - SM0,DM1 - N:C = 0111
10240                                 ;;*********************************************************************
10241  035346                         TST525:
10242  035346  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10243  035350  012700  000525                 MOV     #525,R0         ;;LOAD R0 WITH TEST NUMBER
10244  035354  013701  035404                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10245  035360  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10246  035364  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
10247  035370  012705  125252                 MOV     #125252,R5      ;SRC OPR = 125252
10248  035374  012712  052525                 MOV     #52525,(R2)     ;[DEST] = 52525
10249  035400  000257                         CCC                     ;CLEAR FLAGS
10250  035402  000267                         267                     ;N:C = 0111
10251
10252  035404  050512                 2$:     BIS     R5,(R2)         ;TEST THE BIS
10253
10254  035406  100003                         BPL     3$              ;N:C = 1001
10255  035410  001402                         BEQ     3$
10256  035412  102401                         BVS     3$
10257  035414  103401                         BCS     4$
10258
10259  035416  104001                 3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10260
10261  035420  020412                 4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10262  035422  001402                         BEQ     TST526          ;;BR IF YES
```

```
10263
10264  035424  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10265  035426  104001            5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10266
10267                            ;;*****************************************************************
10268                            ;*TEST 526      BIS TEST - SM0,DM1 - N:C = 1000
10269                            ;;*****************************************************************
10270  035430                    TST526:
10271  035430  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10272  035432  012700  000526            MOV     #526,R0         ;;LOAD R0 WITH TEST NUMBER
10273  035436  013701  035460            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10274  035442  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10275  035446  005004                    CLR     R4              ;RESULT S / B = 000000
10276  035450  005005                    CLR     R5              ;SRC OPR = 000000
10277  035452  005012                    CLR     (R2)            ;[DEST] = 000000
10278  035454  000257                    CCC                     ;CLEAR FLAGS
10279  035456  000270                    SEN                     ;N:C = 1000
10280
10281  035460  050512            2$:     BIS     R5,(R2)         ;TEST THE BIS
10282
10283  035462  100403                    BMI     3$              ;N:C = 0100
10284  035464  001002                    BNE     3$
10285  035466  102401                    BVS     3$
10286  035470  103001                    BCC     4$
10287
10288  035472  104001            3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10289
10290  035474  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10291  035476  001402                    BEQ     TST527          ;;BR IF YES
10292
10293  035500  011203                    MOV     (R2),R3         ;GET THE WAS DATA
10294  035502  104001            5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10295
10296                            ;;*****************************************************************
10297                            ;*TEST 527      BIC TEST - SM0,DM1 - N:C = 0111
10298                            ;;*****************************************************************
10299  035504                    TST527:
10300  035504  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10301  035506  012700  000527            MOV     #527,R0         ;;LOAD R0 WITH TEST NUMBER
10302  035512  013701  035542            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10303  035516  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10304  035522  012704  100000            MOV     #100000,R4      ;RESULT S / B = 100000
10305  035526  012705  077777            MOV     #77777,R5               ;SRC OPR = 77777
10306  035532  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
10307  035536  000257                    CCC                     ;CLEAR FLAGS
10308  035540  000267                    267                     ;N:C = 0111
10309
10310  035542  040512            2$:     BIC     R5,(R2)         ,TEST THE BIC
10311
10312  035544  100003                    BPL     3$              ;N:C = 1001
10313  035546  001402                    BEQ     3$
10314  035550  102401                    BVS     3$
10315  035552  103401                    BCS     4$
10316
10317  035554  104001            3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10318
```

```
10319  035556  020412                4$:   CMP   R4,(R2)          ;CORRECT RESULT ?
10320  035560  001402                      BEQ   TST530           ;;BR IF YES
10321
10322  035562  011203                      MOV   (R2),R3          ;GET THE WAS DATA
10323  035564  104001                5$:   ERROR 1                ;BIC DELIVERED THE WRONG RESULT
10324
10325                                ;;*****************************************************************
10326                                ;*TEST 530      BIC TEST - SM0,DM1 - N:C = 1000
10327                                ;;*****************************************************************
10328  035566                        TST530:
10329  035566  000004                      SCOPE                  ;CALL THE SCOPE LOOP UTILITY
10330  035570  012700  000530              MOV   #530,R0          ;;LOAD R0 WITH TEST NUMBER
10331  035574  013701  035616              MOV   @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
10332  035600  012702  063312              MOV   #MBUF0,R2        ;DEST ADDR = MBUF0
10333  035604  005004                      CLR   R4               ;RESULT S / B = 000000
10334  035606  005005                      CLR   R5               ;SRC OPR = 000000
10335  035610  005012                      CLR   (R2)             ;[DEST] = 000000
10336  035612  000257                      CCC                    ;CLEAR FLAGS
10337  035614  000270                      SEN                    ;N:C = 1000
10338
10339  035616  040512                2$:   BIC   R5,(R2)          ;TEST THE BIC
10340
10341  035620  100403                      BMI   3$               ;N:C = 0100
10342  035622  001002                      BNE   3$
10343  035624  102401                      BVS   3$
10344  035626  103001                      BCC   4$
10345
10346  035630  104001                3$:   ERROR 1                ;BIC FAILED TO ALTER CODES PROPERLY
10347
10348  035632  020412                4$:   CMP   R4,(R2)          ;CORRECT RESULT ?
10349  035634  001402                      BEQ   TST531           ;;BR IF YES
10350
10351  035636  011203                      MOV   (R2),R3          ;GET THE WAS DATA
10352  035640  104001                5$:   ERROR 1                ;BIC DELIVERED THE WRONG RESULT
10353
10354                                ;;*****************************************************************
10355                                ;*TEST 531      BIT TEST - SM0,DM1 - N:C = 0111
10356                                ;;*****************************************************************
10357  035642                        TST531:
10358  035642  000004                      SCOPE                  ;CALL THE SCOPE LOOP UTILITY
10359  035644  012700  000531              MOV   #531,R0          ;;LOAD R0 WITH TEST NUMBER
10360  035650  013701  035700              MOV   @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
10361  035654  012702  063312              MOV   #MBUF0,R2        ;DEST ADDR = MBUF0
10362  035660  012704  100000              MOV   #100000,R4       ;RESULT S / B = 100000
10363  035664  012705  100000              MOV   #100000,R5       ;SRC OPR = 100000
10364  035670  012712  100000              MOV   #100000,(R2)     ;[DEST] = 100000
10365  035674  000257                      CCC                    ;CLEAR FLAGS
10366  035676  000267                      267                    ;N:C = 0111
10367
10368  035700  030512                2$:   BIT   R5,(R2)          ;TEST THE BIT
10369
10370  035702  100003                      BPL   3$               ;N:C = 1001
10371  035704  001402                      BEQ   3$
10372  035706  102401                      BVS   3$
10373  035710  103401                      BCS   4$
10374
```

```
10375  035712  104001              3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10376
10377  035714  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10378  035716  001402                      BEQ     TST532          ;;BR IF YES
10379
10380  035720  011203                      MOV     (R2),R3         ;GET THE WAS DATA
10381  035722  104001              5$:     ERROR   1               ;BIT DELIVERED A RESULT
10382
10383                              ;;**********************************************************************
10384                              ;*TEST 532       BIT TEST - SM0,DM1 - N:C = 1000
10385                              ;;**********************************************************************
10386  035724                      TST532:
10387  035724  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10388  035726  012700  000532              MOV     #532,R0         ;;LOAD R0 WITH TEST NUMBER
10389  035732  013701  035762              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10390  035736  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10391  035742  012704  052525              MOV     #52525,R4       ;RESULT S / B = 52525
10392  035746  012705  125252              MOV     #125252,R5      ;SRC OPR = 125252
10393  035752  012712  052525              MOV     #52525,(R2)     ;[DEST] = 52525
10394  035756  000257                      CCC                     ;CLEAR FLAGS
10395  035760  000270                      SEN                     ;N:C = 1000
10396
10397  035762  030512              2$:     BIT     R5,(R2)         ;TEST THE BIT
10398
10399  035764  100403                      BMI     3$              ;N:C = 0100
10400  035766  001002                      BNE     3$
10401  035770  102401                      BVS     3$
10402  035772  103001                      BCC     4$
10403
10404  035774  104001              3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10405
10406  035776  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10407  036000  001402                      BEQ     TST533          ;;BR IF YES
10408
10409  036002  011203                      MOV     (R2),R3         ;GET THE WAS DATA
10410  036004  104001              5$:     ERROR   1               ;BIT DELIVERED A RESULT
10411                              ;;**********************************************************************
10412                              ;*TEST 533       CMP TEST - SM0,DM1 - N:C = 1010
10413                              ;;**********************************************************************
10414  036006                      TST533:
10415  036006  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10416  036010  012700  000533              MOV     #533,R0         ;;LOAD R0 WITH TEST NUMBER
10417  036014  013701  036044              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10418  036020  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10419  036024  012704  177777              MOV     #-1,R4          ;RESULT S / B = -1
10420  036030  012705  177777              MOV     #-1,R5          ;SRC OPR = 177777
10421  036034  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
10422  036040  000257                      CCC                     ;CLEAR FLAGS
10423  036042  000272                      272                     ;N:C = 1010
10424
10425  036044  020512              2$:     CMP     R5,(R2)         ;TEST THE CMP
10426
10427  036046  100403                      BMI     3$              ;N:C = 0100
10428  036050  001002                      BNE     3$
10429  036052  102401                      BVS     3$
10430  036054  103001                      BCC     4$
```

```
10431
10432  036056  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10433
10434  036060  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10435  036062  001402                          BEQ     TST534          ;;BR IF YES
10436
10437  036064  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10438  036066  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10439
10440                                  ;;**********************************************************************
10441                                  ;*TEST 534       CMP TEST - SM0,DM1 - N:C = 0110
10442                                  ;;**********************************************************************
10443  036070                          TST534:
10444  036070  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10445  036072  012700  000534                  MOV     #534,R0         ;;LOAD R0 WITH TEST NUMBER
10446  036076  013701  036124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10447  036102  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10448  036106  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10449  036112  005005                          CLR     R5              ;SRC OPR = 000000
10450  036114  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10451  036120  000257                          CCC                     ;CLEAR FLAGS
10452  036122  000266                          266                     ;N:C = 0110
10453
10454  036124  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10455
10456  036126  100003                          BPL     3$              ;N:C = 1001
10457  036130  001402                          BEQ     3$
10458  036132  102401                          BVS     3$
10459  036134  103401                          BCS     4$
10460
10461  036136  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10462
10463  036140  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10464  036142  001402                          BEQ     TST535          ;;BR IF YES
10465
10466  036144  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10467  036146  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10468
10469                                  ;;**********************************************************************
10470                                  ;*TEST 535       CMP TEST - SM0,DM1 - N:C = 0000
10471                                  ;;**********************************************************************
10472  036150                          TST535:
10473  036150  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10474  036152  012700  000535                  MOV     #535,R0         ;;LOAD R0 WITH TEST NUMBER
10475  036156  013701  036204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10476  036162  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10477  036166  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10478  036172  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10479  036176  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10480  036202  000257                          CCC                     ;CLEAR FLAGS
10481
10482  036204  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10483
10484  036206  100403                          BMI     3$              ;N:C = 0010
10485  036210  001402                          BEQ     3$
10486  036212  102001                          BVC     3$
```

```
10487  036214  103001                  BCC     4$
10488
10489  036216  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10490
10491  036220  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10492  036222  001402                  BEQ     TST536          ;;BR IF YES
10493
10494  036224  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10495  036226  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10496
10497                          ;;**********************************************************************
10498                          ;*TEST 536      BIS TEST - SM1,DM0 - N:C = 0111
10499                          ;;**********************************************************************
10500  036230                  TST536:
10501  036230  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10502  036232  012700  000536          MOV     #536,R0         ;;LOAD R0 WITH TEST NUMBER
10503  036236  013701  036262          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10504  036242  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
10505  036246  012705  063332          MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10506  036252  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
10507  036256  000257                  CCC                     ;CLEAR FLAGS
10508  036260  000267                  267                     ;N:C = 0111
10509
10510  036262  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10511
10512  036264  100003                  BPL     3$              ;N:C = 1001
10513  036266  001402                  BEQ     3$
10514  036270  102401                  BVS     3$
10515  036272  103401                  BCS     4$
10516
10517  036274  104002          3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10518
10519  036276  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10520  036300  001401                  BEQ     TST537          ;;BR IF YES
10521
10522  036302  104002          5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10523
10524                          ;;**********************************************************************
10525                          ;*TEST 537      BIS TEST - SM1,DM0 - N:C = 1000
10526                          ;;**********************************************************************
10527  036304                  TST537:
10528  036304  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10529  036306  012700  000537          MOV     #537,R0         ;;LOAD R0 WITH TEST NUMBER
10530  036312  013701  036332          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10531  036316  005004                  CLR     R4              ;RESULT S / B =000000
10532  036320  012705  063322          MOV     #DWTA,R5        ;SRC ADDR = DWTA
10533  036324  005003                  CLR     R3              ;[DEST] = 000000
10534  036326  000257                  CCC                     ;CLEAR FLAGS
10535  036330  000270                  SEN                     ;N:C = 1000
10536
10537  036332  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10538
10539  036334  100403                  BMI     3$              ;N:C = 0100
10540  036336  001002                  BNE     3$
10541  036340  102401                  BVS     3$
10542  036342  103001                  BCC     4$
```

```
10543
10544  036344  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10545
10546  036346  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10547  036350  001401                          BEQ     TST540          ;;BR IF YES
10548
10549  036352  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10550
10551                                   ;;*****************************************************************
10552                                   ;*TEST 540       BIC TEST - SM1,DM0 - N:C = 0111
10553                                   ;;*****************************************************************
10554  036354                           TST540:
10555  036354  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10556  036356  012700  000540                  MOV     #540,R0         ;;LOAD R0 WITH TEST NUMBER
10557  036362  013701  036412                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10558  036366  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10559  036372  012705  063316                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10560  036376  012703  177777                  MOV     #-1,R3  ;[DEST] = 177777
10561  036402  012715  077777                  MOV     #77777,(R5)     ;SRC OPR = 77777
10562  036406  000257                          CCC                     ;CLEAR FLAGS
10563  036410  000267                          267                     ;N:C = 0111
10564
10565  036412  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10566
10567  036414  100003                          BPL     3$              ;N:C = 1001 ?
10568  036416  001402                          BEQ     3$
10569  036420  102401                          BVS     3$
10570  036422  103401                          BCS     4$
10571
10572  036424  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10573
10574  036426  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10575  036430  001401                          BEQ     TST541          ;;BR IF YES
10576
10577  036432  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10578
10579                                   ;;*****************************************************************
10580                                   ;*TEST 541       BIC TEST - SM1,DM0 - N:C = 1000
10581                                   ;;*****************************************************************
10582  036434                           TST541:
10583  036434  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10584  036436  012700  000541                  MOV     #541,R0         ;;LOAD R0 WITH TEST NUMBER
10585  036442  013701  036462                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10586  036446  005004                          CLR     R4              ;RESULT S / B = 000000
10587  036450  012705  063322                  MOV     #DWTA,R5        ;SRC ADDR = DWTA
10588  036454  005003                          CLR     R3              ;[DEST] =000000
10589  036456  000257                          CCC                     ;CLEAR FLAGS
10590  036460  000270                          SEN                     ;N:C = 1000
10591
10592  036462  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10593
10594  036464  100403                          BMI     3$              ;N:C = 0100
10595  036466  001002                          BNE     3$
10596  036470  102401                          BVS     3$
10597  036472  103001                          BCC     4$
10598
```

```
10599  036474  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10600
10601  036476  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10602  036500  001401                          BEQ     TST542          ;;BR IF YES
10603
10604  036502  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10605
10606                                  ;;*********************************************************************
10607                                  ;*TEST 542       BIT TEST - SM1,DM0 - N:C = 0111
10608                                  ;;*********************************************************************
10609  036504                          TST542:
10610  036504  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10611  036506  012700  000542                  MOV     #542,R0         ;;LOAD R0 WITH TEST NUMBER
10612  036512  013701  036534                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10613  036516  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10614  036522  012705  063324                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10615  036526  010403                          MOV     R4,R3           ;[DEST] = 100000
10616  036530  000257                          CCC                     ;CLEAR FLAGS
10617  036532  000267                          267                     ;N:C = 0111
10618
10619  036534  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10620
10621  036536  100003                          BPL     3$              ;N:C = 1001 ?
10622  036540  001402                          BEQ     3$
10623  036542  102401                          BVS     3$
10624  036544  103401                          BCS     4$
10625
10626  036546  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10627
10628  036550  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10629  036552  001401                          BEQ     TST543          ;;BR IF YES
10630
10631  036554  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10632
10633                                  ;;*********************************************************************
10634                                  ;*TEST 543       BIT TEST - SM1,DM0 - N:C = 1000
10635                                  ;;*********************************************************************
10636  036556                          TST543:
10637  036556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10638  036560  012700  000543                  MOV     #543,R0         ;;LOAD R0 WITH TEST NUMBER
10639  036564  013701  036606                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10640  036570  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
10641  036574  012705  063332                  MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10642  036600  010403                          MOV     R4,R3           ;[DEST] = 52525
10643  036602  000257                          CCC                     ;CLEAR FLAGS
10644  036604  000270                          SEN                     ;N:C = 1000
10645
10646  036606  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10647
10648  036610  100403                          BMI     3$              ;N:C = 0100
10649  036612  001002                          BNE     3$
10650  036614  102401                          BVS     3$
10651  036616  103001                          BCC     4$
10652
10653  036620  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10654
```

```
10655  036622  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10656  036624  001401                          BEQ     TST544          ;;BR IF YES
10657
10658  036626  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10659                                  ;;**********************************************************************
10660                                  ;*TEST 544       CMP TEST - SM1,DM0 - N:C = 0110
10661                                  ;;**********************************************************************
10662  036630                          TST544:
10663  036630  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10664  036632  012700  000544                  MOV     #544,R0         ;;LOAD R0 WITH TEST NUMBER
10665  036636  013701  036660                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10666  036642  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10667  036646  012705  063322                  MOV     #DWTA,R5        ;SRC ADDR = DWTA
10668  036652  010403                          MOV     R4,R3           ;[DEST] = +1
10669  036654  000257                          CCC                     ;CLEAR FLAGS
10670  036656  000266                          266                     ;N:C = 0110
10671
10672  036660  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10673
10674  036662  100003                          BPL     3$              ;N:C = 1001
10675  036664  001402                          BEQ     3$
10676  036666  102401                          BVS     3$
10677  036670  103401                          BCS     4$
10678
10679  036672  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10680
10681  036674  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10682  036676  001401                          BEQ     TST545          ;;BR IF YES
10683
10684  036700  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10685
10686                                  ;;**********************************************************************
10687                                  ;*TEST 545       CMP TEST - SM1,DM0 - N:C = 1010
10688                                  ;;**********************************************************************
10689  036702                          TST545:
10690  036702  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10691  036704  012700  000545                  MOV     #545,R0         ;;LOAD R0 WITH TEST NUMBER
10692  036710  013701  036732                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10693  036714  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10694  036720  012705  063324                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10695  036724  010403                          MOV     R4,R3           ;[DEST] = 177777
10696  036726  000257                          CCC                     ;CLEAR FLAGS
10697  036730  000272                          272                     ;N:C = 1010
10698
10699  036732  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10700
10701  036734  100403                          BMI     3$              ;N:C = 0100
10702  036736  001002                          BNE     3$
10703  036740  102401                          BVS     3$
10704  036742  103001                          BCC     4$
10705
10706  036744  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10707
10708  036746  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10709  036750  001401                          BEQ     TST546          ;;BR IF YES
10710
```

```
10711  036752  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10712
10713                                  ;;*****************************************************************
10714                                  ;*TEST 546       CMP TEST - SM1,DM0 - N:C = 0000
10715                                  ;;*****************************************************************
10716  036754                          TST546:
10717  036754  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10718  036756  012700  000546                  MOV     #546,R0         ;;LOAD R0 WITH TEST NUMBER
10719  036762  013701  037010                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10720  036766  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10721  036772  012705  063316                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10722  036776  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10723  037002  012715  100000                  MOV     #100000,(R5)    ;SRC OPR = 100000
10724  037006  000257                          CCC                     ;CLEAR FLAGS
10725
10726  037010  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10727
10728  037012  100403                          BMI     3$              ;N:C = 0010
10729  037014  001402                          BEQ     3$
10730  037016  102001                          BVC     3$
10731  037020  103001                          BCC     4$
10732
10733  037022  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10734
10735  037024  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10736  037026  001401                          BEQ     TST547          ;;BR IF YES
10737
10738  037030  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10739
10740                                  ;;*****************************************************************
10741                                  ;*TEST 547       BIS  SM1,DM1 TEST - N:C = 0111
10742                                  ;;*****************************************************************
10743  037032                          TST547:
10744  037032  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10745  037034  012700  000547                  MOV     #547,R0         ;;LOAD R0 WITH TEST NUMBER
10746  037040  013701  037070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10747  037044  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10748  037050  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 1777777
10749  037054  012705  063332                  MOV     #DWTA+10,R5     ;SOURCE ADDR = DWTA+10
10750  037060  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 052525
10751  037064  000257                          CCC                     ;CLEAR FLAGS
10752  037066  000267                          267                     ;N:C = 0111
10753
10754  037070  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10755
10756  037072  100003                          BPL     3$              ;N:C = 1001?
10757  037074  001402                          BEQ     3$
10758  037076  102401                          BVS     3$
10759  037100  103401                          BCS     4$
10760
10761  037102  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10762
10763  037104  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10764  037106  001402                          BEQ     TST550          ;;BR IF YES
10765
10766  037110  011203                          MOV     (R2),R3         ;GET THE WAS DATA
```

```
10767  037112  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10768
10769                                  ;;*****************************************************************
10770                                  ;*TEST 550      BIS  SM1,DM1 TEST - N:C = 1000
10771                                  ;;*****************************************************************
10772  037114                          TST550:
10773  037114  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10774  037116  012700  000550                 MOV     #550,R0         ;;LOAD R0 WITH TEST NUMBER
10775  037122  013701  037146                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10776  037126  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10777  037132  005004                         CLR     R4              ;RESULT S / B = 000000
10778  037134  012705  063322                 MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
10779  037140  005012                         CLR     (R2)            ;[DEST] = 000000
10780  037142  000257                         CCC                     ;CLEAR FLAGS
10781  037144  000270                         SEN                     ;N:C = 1000
10782
10783  037146  051512                  2$:    BIS     (R5),(R2)       ;TEST THE BIS
10784
10785  037150  100403                         BMI     3$              ;N:C = 0100 ?
10786  037152  001002                         BNE     3$
10787  037154  102401                         BVS     3$
10788  037156  103001                         BCC     4$
10789
10790  037160  104001                  3$:    ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10791
10792  037162  020412                  4$:    CMP     R4,(R2)         ;CORRECT RESULT ?
10793  037164  001402                         BEQ     TST551          ;;BR IF YES
10794
10795  037166  011203                         MOV     (R2),R3         ;GET THE WAS DATA
10796  037170  104001                  5$:    ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10797
10798                                  ;;*****************************************************************
10799                                  ;*TEST 551      BIC  SM1,DM1 TEST - N:C = 0111
10800                                  ;;*****************************************************************
10801  037172                          TST551:
10802  037172  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10803  037174  012700  000551                 MOV     #551,R0         ;;LOAD R0 WITH TEST NUMBER
10804  037200  013701  037234                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10805  037204  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10806  037210  012704  100000                 MOV     #100000,R4      ;RESULT S / B = 100000
10807  037214  012705  063316                 MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10808  037220  012715  077777                 MOV     #77777,(R5)     ;[SOURCE] = 77777
10809  037224  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
10810  037230  000257                         CCC                     ;CLEAR FLAGS
10811  037232  000267                         267                     ;N:C = 0111
10812
10813  037234  041512                  2$:    BIC     (R5),(R2)       ;TEST THE BIC
10814
10815  037236  100003                         BPL     3$              ;N:C = 1001 ?
10816  037240  001402                         BEQ     3$
10817  037242  102401                         BVS     3$
10818  037244  103401                         BCS     4$
10819
10820  037246  104001                  3$:    ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10821
10822  037250  020412                  4$:    CMP     R4,(R2)         ;CORRECT RESULT ?
```

```
10823  037252  001402                  BEQ     TST552          ;;BR IF YES
10824
10825  037254  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10826  037256  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10827
10828                          ;;**********************************************************************
10829                          ;*TEST 552       BIC  SM1,DM1 TEST - N:C = 1000
10830                          ;;**********************************************************************
10831  037260                  TST552:
10832  037260  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10833  037262  012700  000552          MOV     #552,R0         ;;LOAD R0 WITH TEST NUMBER
10834  037266  013701  037314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10835  037272  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10836  037276  005004                  CLR     R4              ;RESULT S / B = 000000
10837  037300  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10838  037304  005015                  CLR     (R5)            ;[SOURCE] = 000000
10839  037306  005012                  CLR     (R2)            ;[DEST] = 000000
10840  037310  000257                  CCC                     ;CLEAR FLAGS
10841  037312  000270                  SEN                     ;N:C = 1000
10842
10843  037314  041512          2$:     BIC     (R5),(R2)       ;TEST THE BIC
10844
10845  037316  100403                  BMI     3$              ;N:C = 0100 ?
10846  037320  001002                  BNE     3$
10847  037322  102401                  BVS     3$
10848  037324  103001                  BCC     4$
10849
10850  037326  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10851
10852  037330  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10853  037332  001402                  BEQ     TST553          ;;BR IF YES
10854
10855  037334  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10856  037336  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10857
10858                          ;;**********************************************************************
10859                          ;*TEST 553       BIT  SM1,DM1 TEST - N:C = 1000
10860                          ;;**********************************************************************
10861  037340                  TST553:
10862  037340  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10863  037342  012700  000553          MOV     #553,R0         ;;LOAD R0 WITH TEST NUMBER
10864  037346  013701  037402          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10865  037352  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10866  037356  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
10867  037362  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10868  037366  012715  052525          MOV     #52525,(R5)     ;[SOURCE] = 052525
10869  037372  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
10870  037376  000257                  CCC                     ;CLEAR FLAGS
10871  037400  000270                  SEN                     ;N:C = 1000
10872
10873  037402  031512          2$:     BIT     (R5),(R2)       ;TEST THE BIT
10874
10875  037404  100403                  BMI     3$              ;N:C = 0100 ?
10876  037406  001002                  BNE     3$
10877  037410  102401                  BVS     3$
10878  037412  103001                  BCC     4$
```

```
10879
10880  037414  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10881
10882  037416  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10883  037420  001402                          BEQ     TST554          ;;BR IF YES
10884
10885  037422  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10886  037424  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10887
10888                                  ;;**********************************************************************
10889                                  ;*TEST 554       BIT  SM1,DM1 TEST - N:C = 0111
10890                                  ;;**********************************************************************
10891  037426                          TST554:
10892  037426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10893  037430  012700  000554                  MOV     #554,R0         ;;LOAD R0 WITH TEST NUMBER
10894  037434  013701  037502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10895                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT11
10896  037440  032737  004000  063234          BIT     #BIT11,@#BPTLOC ;BREAKPOINT HALT SET ??
10897  037446  001401                          BEQ     .+4             ;BR IF NOT
10898  037450  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10899  037452  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10900  037456  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10901  037462  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10902  037466  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 100000
10903  037472  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10904  037476  000257                          CCC                     ;CLEAR FLAGS
10905  037500  000267                          267                     ;N:C = 0111
10906
10907  037502  031512                  2$:     BIT     (R5),(R2)       ;TEST THE BIT
10908
10909  037504  100003                          BPL     3$              ;N:C = 1001 ?
10910  037506  001402                          BEQ     3$
10911  037510  102401                          BVS     3$
10912  037512  103401                          BCS     4$
10913
10914  037514  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10915
10916  037516  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10917  037520  001402                          BEQ     TST555          ;;BR IF YES
10918
10919  037522  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10920  037524  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10921
10922                                  ;;**********************************************************************
10923                                  ;*TEST 555       CMP  SM1,DM1 TEST - N:C = 1010
10924                                  ;;**********************************************************************
10925  037526                          TST555:
10926  037526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10927  037530  012700  000555                  MOV     #555,R0         ;;LOAD R0 WITH TEST NUMBER
10928  037534  013701  037566                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10929  037540  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10930  037544  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10931  037550  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10932  037554  012715  177777                  MOV     #-1,(R5)        ;[SOURCE] = 177777
10933  037560  010412                          MOV     R4,(R2)         ;[DEST] = 177777
10934  037562  000257                          CCC                     ;CLEAR FLAGS
```

```
10935  037564  000272                  272                     ;N:C = 1010
10936
10937  037566  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
10938
10939  037570  100403                  BMI     3$              ;N:C = 0100 ?
10940  037572  001002                  BNE     3$
10941  037574  102401                  BVS     3$
10942  037576  103001                  BCC     4$
10943
10944  037600  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10945
10946  037602  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10947  037604  001402                  BEQ     TST556          ;;BR IF YES
10948
10949  037606  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10950  037610  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10951
10952                          ;;**********************************************************************
10953                          ;*TEST 556       CMP  SM1,DM1 TEST - N:C = 0110
10954                          ;;**********************************************************************
10955  037612                  TST556:
10956  037612  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10957  037614  012700  000556          MOV     #556,R0         ;;LOAD R0 WITH TEST NUMBER
10958  037620  013701  037652          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10959  037624  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10960  037630  012704  000001          MOV     #+1,R4          ;RESULT S / B = 000001
10961  037634  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10962  037640  005015                  CLR     (R5)            ;[SOURCE] = 000000
10963  037642  012712  000001          MOV     #+1,(R2)        ;[DEST] = 000001
10964  037646  000257                  CCC                     ;CLEAR FLAGS
10965  037650  000266                  266                     ;N:C = 0110
10966
10967  037652  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
10968
10969  037654  100003                  BPL     3$              ;N:C = 1001 ?
10970  037656  001402                  BEQ     3$
10971  037660  102401                  BVS     3$
10972  037662  103401                  BCS     4$
10973
10974  037664  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10975
10976  037666  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10977  037670  001402                  BEQ     TST557          ;;BR IF YES
10978
10979  037672  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10980  037674  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10981
10982                          ;;**********************************************************************
10983                          ;*TEST 557       CMP  SM1,DM1 TEST - N:C = 0000
10984                          ;;**********************************************************************
10985  037676                  TST557:
10986  037676  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10987  037700  012700  000557          MOV     #557,R0         ;;LOAD R0 WITH TEST NUMBER
10988  037704  013701  037736          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10989  037710  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10990  037714  012704  000001          MOV     #+1,R4          ;RESULT S / B = 000001
```

```
10991  037720  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10992  037724  012715  100000          MOV     #100000,(R5)    ;[SOURCE] = 000000
10993  037730  012712  000001          MOV     #+1,(R2)        ;[DEST] = 000001
10994  037734  000257                  CCC                     ;CLEAR FLAGS
10995
10996  037736  021512          2$:     CMP     (R5),(R2)       ;TEST THE CMP
10997
10998  037740  100403                  BMI     3$              ;N:C = 0010 ?
10999  037742  001402                  BEQ     3$
11000  037744  102001                  BVC     3$
11001  037746  103001                  BCC     4$
11002
11003  037750  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
11004
11005  037752  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
11006  037754  001402                  BEQ     TST560          ;;BR IF YES
11007
11008  037756  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11009  037760  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
11010
11011                          ;;*****************************************************************
11012                          ;*TEST 560      BISB  SM1,DM0 TEST - SOURCE ADDR ODD
11013                          ;;*****************************************************************
11014  037762                  TST560:
11015  037762  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11016  037764  012700  000560          MOV     #560,R0         ;;LOAD R0 WITH TEST NUMBER
11017  037770  013701  040010          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11018  037774  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11019  040000  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11020  040004  005003                  CLR     R3              ;[DEST] = 000000
11021  040006  000257                  CCC                     ;SCOPE SYNC
11022
11023  040010  151503          2$:     BISB    (R5),R3         ;TEST THE BISB
11024
11025  040012  020403                  CMP     R4,R3           ;RESULT CORRECT ?
11026  040014  001401                  BEQ     TST561          ;;BR IF YES
11027
11028  040016  104002          3$:     ERROR   2               ;BISB DELIVERED THE WRONG RESULT
11029
11030                          ;;*****************************************************************
11031                          ;*TEST 561      BISB  SM1,DM1 TEST - SOURCE ADDR ODD
11032                          ;;*****************************************************************
11033  040020                  TST561:
11034  040020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11035  040022  012700  000561          MOV     #561,R0         ;;LOAD R0 WITH TEST NUMBER
11036  040026  013701  040052          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11037  040032  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11038  040036  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11039  040042  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11040  040046  005012                  CLR     (R2)            ;[DEST] = 000000
11041  040050  000257                  CCC                     ;SCOPE SYNC
11042
11043  040052  151512          2$:     BISB    (R5),(R2)       ;TEST THE BISB
11044
11045  040054  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11046  040056  001402                  BEQ     TST562          ;;BR IF YES
```

```
11047
11048  040060  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11049  040062  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11050
11051                            ;;*****************************************************************
11052                            ;*TEST 562       BISB  SM1,DM2 TEST - SOURCE ADDR ODD
11053                            ;;*****************************************************************
11054  040064                    TST562:
11055  040064  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11056  040066  012700  000562           MOV     #562,R0         ;;LOAD R0 WITH TEST NUMBER
11057  040072  013701  040120           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11058  040076  012702  063312           MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11059  040102  012704  000377           MOV     #377,R4         ;RESULT S / B = 377
11060  040106  012705  064631           MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11061  040112  005012                   CLR     (R2)            ;[DEST] = 000000
11062  040114  010203                   MOV     R2,R3           ;DEST ADDR IN R3
11063  040116  000257                   CCC                     ;SCOPE SYNC
11064
11065  040120  151523            2$:     BISB    (R5),(R3)+      ;TEST THE BISB
11066
11067  040122  020412                   CMP     R4,(R2)         ;CORRECT RESULT
11068  040124  001402                   BEQ     TST563          ;;BR IF YES
11069
11070  040126  011203                   MOV     (R2),R3         ;GET THE WAS DATA
11071  040130  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11072
11073                            ;;*****************************************************************
11074                            ;*TEST 563       BISB  SM1,DM3 TEST - SOURCE ADDR ODD
11075                            ;;*****************************************************************
11076  040132                    TST563:
11077  040132  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11078  040134  012700  000563           MOV     #563,R0         ;;LOAD R0 WITH TEST NUMBER
11079  040140  013701  040170           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11080  040144  012702  063312           MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11081  040150  012704  000377           MOV     #377,R4         ;RESULT S / B = 377
11082  040154  012705  064631           MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11083  040160  005012                   CLR     (R2)            ;[DEST] = 000000
11084  040162  012703  063306           MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
11085  040166  000257                   CCC                     ;SCOPE SYNC
11086
11087  040170  151533            2$:     BISB    (R5),@(R3)+     ;TEST THE BISB
11088
11089  040172  020412                   CMP     R4,(R2)         ;CORRECT RESULT
11090  040174  001402                   BEQ     TST564          ;;BR IF YES
11091
11092  040176  011203                   MOV     (R2),R3         ;GET THE WAS DATA
11093  040200  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11094
11095                            ;;*****************************************************************
11096                            ;*TEST 564       BISB  SM1,DM4 TEST - SOURCE ADDR ODD
11097                            ;;*****************************************************************
11098  040202                    TST564:
11099  040202  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11100  040204  012700  000564           MOV     #564,R0         ;;LOAD R0 WITH TEST NUMBER
11101  040210  013701  040240           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11102  040214  012702  063312           MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
11103  040220  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
11104  040224  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11105  040230  012703  063314          MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
11106  040234  005012                  CLR     (R2)            ;[DEST] = 000000
11107  040236  000257                  CCC                     ;SCOPE SYNC
11108
11109  040240  151543          2$:     BISB    (R5),-(R3)      ;TEST THE BISB
11110
11111  040242  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11112  040244  001402                  BEQ     TST565          ;;BR IF YES
11113
11114  040246  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11115  040250  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11116
11117                          ;;**********************************************************************
11118                          ;*TEST 565      BISB  SM1,DM5 TEST - SOURCE ADDR ODD
11119                          ;;**********************************************************************
11120  040252                  TST565:
11121  040252  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11122  040254  012700  000565          MOV     #565,R0         ;;LOAD R0 WITH TEST NUMBER
11123  040260  013701  040310          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11124  040264  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11125  040270  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11126  040274  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11127  040300  012703  063310          MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
11128  040304  005012                  CLR     (R2)            ;[DEST] = 000000
11129  040306  000257                  CCC                     ;SCOPE SYNC
11130
11131  040310  151553          2$:     BISB    (R5),@-(R3)     ;TEST THE BISB
11132
11133  040312  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11134  040314  001402                  BEQ     TST566          ;;BR IF YES
11135
11136  040316  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11137  040320  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11138
11139                          ;;**********************************************************************
11140                          ;*TEST 566      BISB  SM1,DM6 TEST - SOURCE ADDR ODD
11141                          ;;**********************************************************************
11142  040322                  TST566:
11143  040322  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11144  040324  012700  000566          MOV     #566,R0         ;;LOAD R0 WITH TEST NUMBER
11145  040330  013701  040360          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11146  040334  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11147  040340  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11148  040344  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11149  040350  012703  063320          MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
11150  040354  005012                  CLR     (R2)            ;[DEST] = 000000
11151  040356  000257                  CCC                     ;SCOPE SYNC
11152
11153  040360  151563  177772  2$:     BISB    (R5),-6(R3)     ;TEST THE BISB
11154
11155  040364  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11156  040366  001402                  BEQ     TST567          ;;BR IF YES
11157
11158  040370  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
11159  040372  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11160
11161                                ;;*****************************************************************
11162                                ;*TEST 567      BISB  SM1,DM7 TEST - SOURCE ADDR ODD
11163                                ;;*****************************************************************
11164  040374                        TST567:
11165  040374  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11166  040376  012700  000567               MOV     #567,R0         ;;LOAD R0 WITH TEST NUMBER
11167  040402  013701  040432               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11168  040406  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11169  040412  012704  000377               MOV     #377,R4         ;RESULT S / B = 377
11170  040416  012705  064631               MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11171  040422  012703  063276               MOV     #ATA,R3         ;BASE DEST ADDR = ATA
11172  040426  005012                        CLR     (R2)            ;[DEST] = 000000
11173  040430  000257                        CCC                     ;SCOPE SYNC
11174
11175  040432  151573  000010        2$:     BISB    (R5),@10(R3)    ;TEST THE BISB
11176
11177  040436  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11178  040440  001402                        BEQ     TST570          ;;BR IF YES
11179
11180  040442  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11181  040444  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11182
11183                                ;;*****************************************************************
11184                                ;*TEST 570      BISB  SM0,DM2 TEST - DEST ADDR EVEN
11185                                ;;*****************************************************************
11186  040446                        TST570:
11187  040446  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11188  040450  012700  000570               MOV     #570,R0         ;;LOAD R0 WITH TEST NUMBER
11189  040454  013701  040476               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11190  040460  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11191  040464  012704  000377               MOV     #377,R4         ;RESULT S / B = 377
11192  040470  010203                        MOV     R2,R3           ;DEST ADDR IN R3
11193  040472  005012                        CLR     (R2)            ;[DEST] = 000000
11194  040474  000257                        CCC                     ;SCOPE SYNC
11195
11196  040476  150423                2$:     BISB    R4,(R3)+        ;TEST THE BISB
11197
11198  040500  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11199  040502  001402                        BEQ     TST571          ;;BR IF YES
11200
11201  040504  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11202  040506  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11203
11204                                ;;*****************************************************************
11205                                ;*TEST 571      BISB  SM0,DM1 TEST - DEST ADDR ODD
11206                                ;;*****************************************************************
11207  040510                        TST571:
11208  040510  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11209  040512  012700  000571               MOV     #571,R0         ;;LOAD R0 WITH TEST NUMBER
11210  040516  013701  040546               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11211  040522  012702  063312               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11212  040526  012704  177400               MOV     #177400,R4      ;RESULT S / B = 177400
11213  040532  012705  000377               MOV     #377,R5         ;[R5]=SOURCE OPR = 377
11214  040536  012703  063313               MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
```

```
11215  040542  005012                    CLR     (R2)            ;[DEST] = 000000
11216  040544  000257                    CCC                     ;SCOPE SYNC
11217
11218  040546  150513            2$:     BISB    R5,(R3)         ;TEST THE BISB
11219
11220  040550  020412                    CMP     R4,(R2)         ;CORRECT RESULT
11221  040552  001402                    BEQ     TST572          ;;BR IF YES
11222
11223  040554  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11224  040556  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11225
11226                            ;;**********************************************************************
11227                            ;*TEST 572      BISB  SM0,DM1 TEST - DEST ADDR EVEN
11228                            ;;**********************************************************************
11229  040560                    TST572:
11230  040560  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11231  040562  012700  000572            MOV     #572,R0         ;;LOAD R0 WITH TEST NUMBER
11232  040566  013701  040610            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11233  040572  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11234  040576  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
11235  040602  010203                    MOV     R2,R3           ;DEST ADDR IN R3
11236  040604  005012                    CLR     (R2)            ;[DEST] = 000000
11237  040606  000257                    CCC                     ;SCOPE SYNC
11238
11239  040610  150413            2$:     BISB    R4,(R3)         ;TEST THE BISB
11240
11241  040612  020412                    CMP     R4,(R2)         ;CORRECT RESULT
11242  040614  001402                    BEQ     TST573          ;;BR IF YES
11243
11244  040616  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11245  040620  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11246
11247                            ;;**********************************************************************
11248                            ;*TEST 573      BISB  SM1,DM1 TEST - DEST ADDR ODD
11249                            ;;**********************************************************************
11250  040622                    TST573:
11251  040622  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11252  040624  012700  000573            MOV     #573,R0         ;;LOAD R0 WITH TEST NUMBER
11253  040630  013701  040660            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11254  040634  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11255  040640  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
11256  040644  012705  064631            MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11257  040650  012703  063313            MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
11258  040654  005012                    CLR     (R2)            ;[DEST] = 000000
11259  040656  000257                    CCC                     ;SCOPE SYNC
11260
11261  040660  151513            2$:     BISB    (R5),(R3)       ;TEST THE BISB
11262
11263  040662  020412                    CMP     R4,(R2)         ;CORRECT RESULT
11264  040664  001402                    BEQ     TST574          ;;BR IF YES
11265
11266  040666  011203                    MOV     (R2),R3         ;GET THE WAS DATA
11267  040670  104001            3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11268
11269                            ;;**********************************************************************
11270                            ;*TEST 574      JMP MODE 1 TEST, FLAGS = 1111
```

```
11271
11272  040672
11273  040672  000004
11274  040674  012700  000574          ;;**********************************************************************
11275  040700  013701  040712          TST574:
11276  040704  012702  040720                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11277  040710  000277                          MOV     #574,R0         ;;LOAD R0 WITH TEST NUMBER
11278                                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11279  040712  000112                          MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11280                                          SCC                     ;MAKE N:C = 1111
11281  040714  104006                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11282  040716  000405
11283                                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11284  040720  103003                          BR      TST575          ;;GO CALL SCOPE
11285  040722  102002
11286  040724  001001                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11287  040726  100401                          BVC     5$              ;BR IF JMP CLEARED "V"
11288                                          BNE     5$              ;BR IF JMP CLEARED "Z"
11289  040730  104006                          BMI     TST575          ;;BR IF "N" STILL SET
11290
11291                                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED ONE
11292
11293                                  ;;**********************************************************************
11294  040732                          ;*TEST 575       JMP MODE 1 TEST, FLAGS = 0000
11295  040732  000004                  ;;**********************************************************************
11296  040734  012700  000575          TST575:
11297  040740  013701  040752                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11298  040744  012702  040760          MOV     #575,R0         ;;LOAD R0 WITH TEST NUMBER
11299  040750  000257                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11300                                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11301  040752  000112                  CCC                     ;MAKE N:C = 0000
11302                          2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11303  040754  104006
11304  040756  000405          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11305                                  BR      TST576          ;;GO CALL SCOPE
11306  040760  103403
11307  040762  102402          4$:     BCS     5$              ;BR IF JMP SET "C"
11308  040764  001401                  BVS     5$              ;BR IF JMP SET "V"
11309  040766  100001                  BEQ     5$              ;BR IF JMP SET "Z"
11310                                  BPL     TST576          ;;BR IF "N" STILL CLEAR
11311  040770  104006
11312                          5$:     ERROR   6               ;JMP ALTERED CODES - SET ONE
11313
11314                          ;;**********************************************************************
11315                          ;*TEST 576       JMP MODE 2 TEST; FLAGS = 1111
11316  040772                  ;;**********************************************************************
11317  040772  000004          TST576:
11318  040774  012700  000576          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11319  041000  013701  041012          MOV     #576,R0         ;;LOAD R0 WITH TEST NUMBER
11320  041004  012702  041020          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11321  041010  000277                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11322                                  SCC                     ;SET N:C = 1111
11323  041012  000122
11324                          2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11325  041014  104006
11326  041016  000411          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
                                        BR      TST577          ;;GO TO SCOPE EXIT
```

```
11327
11328  041020  103003          4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11329  041022  102002                  BVC     5$              ;BR IF JMP CLEARED "V"
11330  041024  001001                  BNE     5$              ;BR IF JMP CLEARED "Z"
11331  041026  100401                  BMI     6$              ;BR IF "N" STILL SET
11332
11333  041030  104006          5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11334
11335  041032  022702  041022  6$:     CMP     #4$+2,R2        ;DID R2 GET AUTO-INCREMENTED?
11336  041036  001401                  BEQ     TST577          ;;BR IF YES
11337
11338  041040  104006          7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER (R2)
11339
11340                          ;;**************************************************************************
11341                          ;*TEST 577       JMP MODE 2 TEST; FLAGS = 0000
11342                          ;;**************************************************************************
11343  041042                  TST577:
11344  041042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11345  041044  012700  000577          MOV     #577,R0         ;;LOAD R0 WITH TEST NUMBER
11346  041050  013701  041062          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11347  041054  012702  041070          MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11348  041060  000257                  CCC                     ;MAKE N:C = 0000
11349
11350  041062  000122          2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11351
11352  041064  104006          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11353  041066  000405                  BR      TST600          ;;GO TO SCOPE EXIT
11354
11355  041070  103403          4$:     BCS     5$              ;BR IF JMP SET "C"
11356  041072  102402                  BVS     5$              ;BR IF JMP SET "V"
11357  041074  001401                  BEQ     5$              ;BR IF JMP SET "Z"
11358  041076  100001                  BPL     TST600          ;;BR IF "N" IS CLEAR
11359
11360  041100  104006          5$:     ERROR   6               ;JMP ALTERED CODES - SET
11361
11362                          ;;**************************************************************************
11363                          ;*TEST 600       JMP TEST MODE 3; FLAGS = 1111
11364                          ;;**************************************************************************
11365  041102                  TST600:
11366  041102  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11367  041104  012700  000600          MOV     #600,R0         ;;LOAD R0 WITH TEST NUMBER
11368  041110  013701  041122          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11369  041114  012702  041154          MOV     #7$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11370  041120  000277                  SCC                     ;SET N:C = 1111
11371
11372  041122  000132          2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11373
11374  041124  104006          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11375  041126  000414                  BR      TST601          ;;GO TO SCOPE EXIT
11376
11377  041130  103003          4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11378  041132  102002                  BVC     5$              ;BR IF JMP CLEARED "V"
11379  041134  001001                  BNE     5$              ;BR IF JMP CLEARED "Z"
11380  041136  100401                  BMI     6$              ;BR IF "N" STILL SET
11381
11382  041140  104006          5$:     ERROR   6               ;JMP ALTERED CODES - CLEAR
```

```
 11383
 11384  041142  022702  041156          6$:     CMP     #7$+2,R2        ;DID JMP UPDATE R2?
 11385  041146  001404                          BEQ     TST601          ;;BR IF YES
 11386
 11387  041150  104006                          ERROR   6               ;JMP FAILED TO UPDATE REGISTER
```

```
11388  041152  000402                    BR      TST601          ;;GO TO SCOPE EXIT
11389  041154  041130            7$:     4$                      ;JMP3 CONTAINS JUMP ADDRESS
11390  041156  104006                    ERROR   6               ;ERROR CALL OCCURS IF MODE3 HAPPENS
11391                                                            ;TO EXECUTE AS MODE 1 OR 2 AND
11392                                                            ;4$ IS LEGAL INSTRUCTION
11393
11394                                    ;;*****************************************************************
11395                                    ;*TEST 601      JMP TEST MODE 3; FLAGS = 0000
11396                                    ;;*****************************************************************
11397  041160                    TST601:
11398  041160  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11399  041162  012700  000601            MOV     #601,R0         ;;LOAD R0 WITH TEST NUMBER
11400  041166  013701  041200            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11401  041172  012702  041222            MOV     #6$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11402  041176  000257                    CCC                     ;MAKE N:C = 0000
11403
11404  041200  000132            2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11405
11406  041202  104006            3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
11407  041204  000410                    BR      TST602          ;;GO TO SCOPE EXIT
11408
11409  041206  103403            4$:     BCS     5$              ;BR IF JMP SET "C"
11410  041210  102402                    BVS     5$              ;BR IF JMP SET "V"
11411  041212  001401                    BEQ     5$              ;BR IF JMP SET "Z"
11412  041214  100004                    BPL     TST602          ;;BR IF "N" STILL CLEAR
11413
11414  041216  104006            5$:     ERROR   6               ;JMP ALTERED CODES - SET
11415  041220  000402                    BR      TST602          ;;GO TO SCOPE EXIT
11416
11417  041222  041206            6$:     4$                      ;JUMP ADDRESS IN 6$
11418  041224  104006                    ERROR   6               ;JMP MODE 3 EXECUTED LIKE MODE 1 OR 2
11419
11420                                    ;;*****************************************************************
11421                                    ;*TEST 602      JMP TEST MODE 4; FLAGS = 1111
11422                                    ;;*****************************************************************
11423  041226                    TST602:
11424  041226  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11425  041230  012700  000602            MOV     #602,R0         ;;LOAD R0 WITH TEST NUMBER
11426  041234  013701  041246            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11427  041240  012702  041256            MOV     #5$,R2  ;[R2] = JMP ADDRESS PLUS 2
11428  041244  000277                    SCC                     ;MAKE N:C = 1111
11429
11430  041246  000142            2$:     JMP     -(R2)           ;TEST THE JMP - GO TO 5$ MINUS 2
11431
11432  041250  104006            3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11433  041252  000414                    BR      TST603          ;;GO TO SCOPE EXIT
11434
11435  041254  000402                    BR      4$              ;GO TEST FLAGS - JMP LOADED PC OK
11436  041256  104006            5$:     ERROR   6               ;JMP FAILED TO AUTO-DECREMENT R2
11437  041260  000411                    BR      TST603          ;;GO TO SCOPE EXIT
11438
11439  041262  103003            4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11440  041264  102002                    BVC     7$              ;BR IF JMP CLEARED "V"
11441  041266  001001                    BNE     7$              ;BR IF JMP CLEARED "Z"
11442  041270  100401                    BMI     6$              ;BR IF "N" STILL SET
11443
```

```
11444  041272  104006                7$:     ERROR   6               ;JMP ALTERED FLAGS
11445
11446  041274  022702  041254        6$:     CMP     #5$-2,R2        ;DID JMP UPDATE R2 PROPERLY?
11447  041300  001401                        BEQ     TST603          ;;BR IF YES
11448
11449  041302  104006                9$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11450
11451                                ;;*****************************************************************
11452                                ;*TEST 603       JMP TEST MODE 4; FLAGS = 0000
11453                                ;;*****************************************************************
11454  041304                        TST603:
11455  041304  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11456  041306  012700  000603                MOV     #603,R0         ;;LOAD R0 WITH TEST NUMBER
11457  041312  013701  041324                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11458  041316  012702  041334                MOV     #4$+2,R2        ;[R2] = JUMP ADDRESS PLUS 2
11459  041322  000257                        CCC                     ;MAKE N:C = 0000
11460
11461  041324  000142                2$:     JMP     -(R2)           ;TEST THE JMP - TO TO 4$
11462
11463  041326  104006                3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11464  041330  000405                        BR      TST604          ;;GO TO SCOPE EXIT
11465
11466  041332  103403                4$:     BCS     5$              ;BR IF JMP SET ''C''
11467  041334  102402                        BVS     5$              ;BR IF JMP SET 'V''
11468  041336  001401                        BEQ     5$              ;BR IF JMP SET 'Z''
11469  041340  100001                        BPL     TST604          ;;BR IF 'N'' STILL CLEAR
11470
11471  041342  104006                5$:     ERROR   6               ;JMP ALTERED CODES - SET
11472
11473                                ;;*****************************************************************
11474                                ;*TEST 604       JMP TEST MODE 5; FLAGS = 1111
11475                                ;;*****************************************************************
11476  041344                        TST604:
11477  041344  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11478  041346  012700  000604                MOV     #604,R0         ;;LOAD R0 WITH TEST NUMBER
11479  041352  013701  041364                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11480  041356  012702  041420                MOV     #JMP5,R2        ;JMP CONTAINS ADDR+2 OF JUMP ADDRESS
11481  041362  000277                        SCC
11482
11483  041364  000152                2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11484
11485  041366  104006                3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11486  041370  000414                        BR      TST605          ;;GO TO SCOPE OXIT
11487
11488  041372  103003                4$:     BCC     5$              ;BR IF JMP CLEARED ''C''
11489  041374  102002                        BVC     5$
11490  041376  001001                        BNE     5$
11491  041400  100401                        BMI     6$
11492
11493  041402  104006                5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11494
11495  041404  022702  041416        6$:     CMP     #JMP5-2,R2      ;DID R2 GET AUTO-DECREMENTED
11496  041410  001404                        BEQ     TST605          ;;BR IF YES
11497
11498  041412  104006                7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
11499  041414  000402                        BR      TST605          ;;GO TO SCOPE EXIT
```

```
11500  041416  041372                        4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11501  041420  104006                JMP5:   ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11502
11503                                  ;;*****************************************************************
11504                                  ;*TEST 605       JMP TEST MODE 5; FLAG = 0000
11505                                  ;;*****************************************************************
11506  041422                          TST605:
11507  041422  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11508  041424  012700  000605               MOV     #605,R0         ;;LOAD R0 WITH TEST NUMBER
11509  041430  013701  041442               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11510  041434  012702  041466               MOV     #JMP5A,R2       ;[R2] = ADDR +2 OF JUMP ADDRESS
11511  041440  000257                        CCC                     ;SET N:C = 0000
11512
11513  041442  000152                2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
11514
11515  041444  104006                3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11516  041446  000410                        BR      TST606          ;;GO TO SCOPE EXIT
11517
11518  041450  103403                4$:     BCS     5$              ;BR IF JMP SET "C"
11519  041452  102402                        BVS     5$              ;BR IF JMP SET "V"
11520  041454  001401                        BEQ     5$              ;BR IF JMP SET "Z"
11521  041456  100004                        BPL     TST606          ;;BR IF "N" STILL CLEAR
11522
11523  041460  104006                5$:     ERROR   6               ;JMP ALTERED THE CODES - SET
11524  041462  000402                        BR      TST606          ;;GO TO SCOPE EXIT
11525
11526  041464  041450                        4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11527  041466  104006                JMP5A:  ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
11528
11529                                  ;;*****************************************************************
11530                                  ;*TEST 606       JMP TEST MODE 6; FLAGS = 1111
11531                                  ;;*****************************************************************
11532  041470                          TST606:
11533  041470  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11534  041472  012700  000606               MOV     #606,R0         ;;LOAD R0 WITH TEST NUMBER
11535  041476  013701  041510               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11536  041502  012702  041534               MOV     #7$,R2  ;[R2] = BASE ADDRESS TO BE INDEXED
11537  041506  000277                        SCC                     ;MAKE N:C = 1111
11538
11539  041510  000162  177764        2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11540
11541  041514  104006                3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
11542  041516  000407                        BR      TST607          ;;GO TO SCOPE EXIT
11543
11544  041520  103003                4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11545  041522  102002                        BVC     5$
11546  041524  001001                        BNE     5$
11547  041526  100403                        BMI     TST607          ;;BR IF "N" STILL SET
11548
11549  041530  104006                5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11550  041532  000401                        BR      TST607          ;;GO TO SCOPE EXIT
11551
11552  041534  104006                7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2 OR
11553                                                                 ;FAILED TO INDEX [R2]
11554
11555                                  ;;*****************************************************************
```

```
11556
11557
11558  041536
11559  041536  000004
11560  041540  012700  000607          ;*TEST 607      JMP TEST MODE 6; FLAGS = 0000
11561  041544  013701  041556          ;;**************************************************************
11562  041550  012702  041602          TST607:
11563  041554  000257                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11564                                           MOV     #607,R0         ;;LOAD R0 WITH TEST NUMBER
11565  041556  000162  177764                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11566                                           MOV     #7$,R2  ;[R2] = BASE ADDRESS FOR JUMP
11567  041562  104006                          CCC                     ;MAKE N:C = 0000
11568  041564  000407
11569                                  2$:      JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11570  041566  103403
11571  041570  102402                  3$:      ERROR   6               ;JMP FAILED TO LOAD PC
11572  041572  001401                          BR      TST610          ;;GO TO SCOPE EXIT
11573  041574  100003
11574                                  4$:      BCS     5$              ;BR IF JMP SET ''C''
11575  041576  104006                          BVS     5$              ;BR IF JMP SET ''V''
11576  041600  000401                          BEQ     5$              ;BR IF JMP SET ''Z''
11577                                           BPL     TST610          ;;BR IF ''N'' STILL CLEAR
11578  041602  104006
11579                                  5$:      ERROR   6               ;JMP ALTERED CODES
11580                                           BR      TST610          ;;GO TO SCOPE EXIT
11581
11582                                  7$:      ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2, OR
11583                                                                   ;FAILED TO INDEX [R2]
11584  041604
11585  041604  000004                  ;;**************************************************************
11586  041606  012700  000610          ;*TEST 610      JMP TEST MODE 7; FLAGS = 1111
11587  041612  013701  041624          ;;**************************************************************
11588  041616  012702  041634          TST610:
11589  041622  000277                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11590                                           MOV     #610,R0         ;;LOAD R0 WITH TEST NUMBER
11591  041624  000172  000020                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11592                                           MOV     #5$,R2  ;[R2] = BASE ADDRESS
11593  041630  104006                          SCC                     ;MAKE N:C = 1111
11594  041632  000412
11595                                  2$:      JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11596  041634  104006
11597  041636  000410                  3$:      ERROR   6               ;JMP FAILED TO LOAD PC
11598                                           BR      TST611          ;;GO TO SCOPE EXIT
11599  041640  103003
11600  041642  102002                  5$:      ERROR   6               ;JMP FAILED TO INDEX OR ACTED LIKE MODE 1 OR 2
11601  041644  001001                          BR      TST611          ;;GO TO SCOPE EXIT
11602  041646  100404
11603                                  4$:      BCC     7$              ;BR IF JMP CLEARED ''C''
11604  041650  104006                          BVC     7$              ;BR IF JMP CLEARED ''V''
11605  041652  000402                          BNE     7$              ;BR IF JMP CLEARED ''Z''
11606                                           BMI     TST611          ;;BR IF ''N'' STILL SET
11607  041654  041640
11608                                  7$:      ERROR   6               ;JMP ALTERED CODES - CLEARED
11609  041656  104006                          BR      TST611          ;;GO TO SCOPE EXIT
11610
11611                                  8$:      4$                      ;THIS LOCATION CONTAINS JMP ADDRESS

                                                ERROR   6               ;JMP EXECUTED LIKE MODE 6

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

```
11612
11613
11614  041660
11615  041660  000004
11616  041662  012700  000611          ;*TEST 611      JMP TEST MODE 7; FLAGS = 0000
11617  041666  013701  041700          ;;*****************************************************************
11618  041672  012702  041710          TST611:
11619  041676  000257                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11620                                           MOV     #611,R0         ;;LOAD R0 WITH TEST NUMBER
11621  041700  000172  000020                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11622                                           MOV     #5$,R2  ;[R2] = BASE ADDRESS
11623  041704  104006                          CCC                     ;MAKE N:C = 0000
11624  041706  000412          2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11625
11626  041710  104006          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11627  041712  000410                  BR      TST612          ;;GO TO SCOPE EXIT
11628
11629  041714  103403          5$:     ERROR   6               ;JMP FAILED TO INDEX
11630  041716  102402                  BR      TST612          ;;GO TO SCOPE EXIT
11631  041720  001401
11632  041722  100004          4$:     BCS     7$              ;BR IF JMP SET ''C''
11633                                   BVS     7$              ;BR IF JMP SET 'V''
11634  041724  104006                  BEQ     7$              ;BR IF JMP SET 'Z''
11635  041726  000402                  BPL     TST612          ;;BR IF 'N'' STILL CLEAR
11636
11637  041730  041714          7$:     ERROR   6               ;JMP ALTERED CODES - SET
11638                                   BR      TST612          ;;GO TO SCOPE EXIT
11639  041732  104006
11640                           8$:     4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11641
11642                                   ERROR   6               ;JMP EXECUTED LIKE A MODE 6
11643
11644  041734                   ;;*****************************************************************
11645  041734  000004           ;*TEST 612      JSR MODE 1 TEST - LOAD PC  /  PUSH SP
11646  041736  012700  000612   ;;*****************************************************************
11647  041742  013701  041764   TST612:
11648  041746  010605                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11649  041750  010737  001010          MOV     #612,R0         ;;LOAD R0 WITH TEST NUMBER
11650  041754  010506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11651  041756  012702  041770          MOV     SP,R5           ;SAVE THE SP
11652  041762  000257                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11653                           1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11654  041764  004412                  MOV     #4$,R2          ;DEST ADDR = 4$
11655                                   CCC                     ;SCOPE SYNC
11656  041766  104006          2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11657
11658  041770  005726          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11659  041772  020605
11660  041774  001406          4$:     TST     (SP)+           ;POP THE SP
11661                                   CMP     SP,R5           ;DID JSR PUSH THE SP ?
11662  041776  005746                  BEQ     TST613          ;;BR IF YES
11663  042000  010603
11664  042002  010504                  TST     -(SP)           ;RESTORE ERROR SP
11665  042004  005744                  MOV     SP,R3           ;[R3]= WAS SP
11666  042006  104003                  MOV     R5,R4
11667                                   TST     -(R4)           ;[R4]= S/B SP
                                5$:     ERROR   3               ;JSR FAILED TO PUSH THE SP
```

```
11668  042010  010506                        MOV     R5,SP           ;RESTORE SP IN CASE OF ERROR
11669                                  ;;*****************************************************************
11670                                  ;*TEST 613      JSR MODE 1 TEST - CHECK RN AND OLD PC
11671                                  ;;*****************************************************************
11672  042012                          TST613:
11673  042012  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11674  042014  012700  000613                MOV     #613,R0         ;;LOAD R0 WITH TEST NUMBER
11675  042020  013701  042052                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11676  042024  010605                        MOV     SP,R5           ;SAVE THE SP
11677  042026  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11678  042032  010506                  1$:   MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11679  042034  012702  042056                MOV     #4$,R2          ;DEST ADDR = 4$
11680  042040  005066  177776                CLR     -2(SP)          ;INIT STACK LOC TO GET [R4]
11681  042044  012704  125252                MOV     #125252,R4      ;INIT RN = 125252
11682  042050  000257                        CCC                     ;SCOPE SYNC
11683
11684  042052  004412                  2$:   JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11685
11686  042054  104006                  3$:   ERROR   6               ;JSR FAILED TO LOAD THE PC
11687
11688  042056  022726  125252          4$:   CMP     #125252,(SP)+   ;DID JSR SAVE REG ON STACK
11689  042062  001401                        BEQ     8$              ;BR IF IT DID
11690
11691  042064  104005                  5$:   ERROR   5               ;JSR FAILED TO SAVE REG ON STACK
11692
11693  042066  022704  042054          8$:   CMP     #3$,R4          ;DID OLD PC GET SAVED ?
11694  042072  001401                        BEQ     6$              ;BR IF YES
11695
11696  042074  104005                  7$:   ERROR   5               ;JSR FAILED TO SAVE TH OLD PC
11697
11698  042076  010506                  6$:   MOV     R5,SP           ;RESTORE SP IN CASE ERROR SCREWED IT UP
11699
11700                                  ;;*****************************************************************
11701                                  ;*TEST 614      JSR MODE 1 TEST - N:C = 0000
11702                                  ;;*****************************************************************
11703  042100                          TST614:
11704  042100  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11705  042102  012700  000614                MOV     #614,R0         ;;LOAD R0 WITH TEST NUMBER
11706  042106  013701  042142                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11707                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT12
11708  042112  032737  010000  063234        BIT     #BIT12,@#BPTLOC ;BREAKPOINT HALT SET ??
11709  042120  001401                        BEQ     .+4             ;BR IF NOT
11710  042122  000000                        HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
11711  042124  010605                        MOV     SP,R5           ;SAVE THE SP
11712  042126  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11713  042132  010506                  1$:   MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11714  042134  012702  042146                MOV     #4$,R2          ;DEST ADDR = 4$
11715  042140  000257                        CCC                     ;N:C = 0000
11716
11717  042142  004412                  2$:   JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11718
11719  042144  104006                  3$:   ERROR   6               ;JSR FAILED TO LOAD THE PC
11720
11721  042146  100403                  4$:   BMI     5$              ;N:C = 0000 ?
11722  042150  001402                        BEQ     5$
11723  042152  102401                        BVS     5$
```

```
11724  042154  103001                  BCC     6$
11725
11726  042156  104005          5$:     ERROR   5               ;JSR FAILED - ALTERED FLAGS
11727
11728  042160  010506          6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11729                          ;;*********************************************************************
11730                          ;*TEST 615       JSR MODE 1 TEST - N:C = 1111
11731                          ;;*********************************************************************
11732  042162                  TST615:
11733  042162  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11734  042164  012700  000615          MOV     #615,R0         ;;LOAD R0 WITH TEST NUMBER
11735  042170  013701  042212          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11736  042174  010605                  MOV     SP,R5           ;SAVE THE SP
11737  042176  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11738  042202  010506          1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11739  042204  012702  042216          MOV     #4$,R2          ;DEST ADDR = 4$
11740  042210  000277                  SCC                     ;N:C = 1111
11741
11742  042212  004412          2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11743
11744  042214  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11745
11746  042216  100003          4$:     BPL     5$              ;N:C = 1111 ?
11747  042220  001002                  BNE     5$
11748  042222  102001                  BVC     5$
11749  042224  103401                  BCS     6$
11750  042226  104005          5$:     ERROR   5               ;JSR ALTERED FLAGS
11751
11752  042230  010506          6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11753
11754                          ;;*********************************************************************
11755                          ;*TEST 616       JSR MODE 2 TEST
11756                          ;;*********************************************************************
11757  042232                  TST616:
11758  042232  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11759  042234  012700  000616          MOV     #616,R0         ;;LOAD R0 WITH TEST NUMBER
11760  042240  013701  042262          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11761  042244  010605                  MOV     SP,R5           ;SAVE THE SP
11762  042246  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11763  042252  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11764  042254  012702  042266          MOV     #4$,R2          ;DEST ADDR = 4$
11765  042260  000257                  CCC                     ;SCOPE SYNC
11766
11767  042262  004422          2$:     JSR     R4,(R2)+        ;TEST THE JSR - GO TO 4$
11768
11769  042264  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11770
11771  042266  005726          4$:     TST     (SP)+           ;RESET SP
11772  042270  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11773  042272  001406                  BEQ     TST617          ;;BR IF YES
11774
11775  042274  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11776  042276  010603                  MOV     SP,R3           ;WAS SP
11777  042300  010504                  MOV     R5,R4
11778  042302  005744                  TST     -(R4)           ;S/B SP
11779  042304  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH SP
```

```
11780
11781  042306  010506                  MOV     R5,SP           ;RESTORE SP JUST IN CASE
11782
11783                                  ;;*****************************************************************
11784                                  ;*TEST 617      JSR MODE 3 TEST
11785                                  ;;*****************************************************************
11786  042310                          TST617:
11787  042310  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11788  042312  012700  000617                 MOV     #617,R0         ;;LOAD R0 WITH TEST NUMBER
11789  042316  013701  042340                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11790  042322  010605                         MOV     SP,R5           ;SAVE THE SP
11791  042324  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11792  042330  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11793  042332  012702  042366                 MOV     #7$,R2          ;DEST ADDR = [7$]
11794  042336  000257                         CCC                     ;SCOPE SYNC
11795
11796  042340  004432                  2$:    JSR     R4,@(R2)+       ;TEST THE JSR - GO TO 4$ VIA 7$
11797
11798  042342  104006                  3$:    ERROR   6               ;JSR FAILED TO LOAD THE PC
11799
11800  042344  005726                  4$:    TST     (SP)+           ;RESET SP
11801  042346  020605                         CMP     SP,R5           ;DID JSR PUSH STACK ?
11802  042350  001411                         BEQ     TST620          ;;BR IF YES
11803
11804  042352  005746                         TST     -(SP)           ;RESET SP TO ERROR VALUE
11805  042354  010603                         MOV     SP,R3           ;WAS SP
11806  042356  010504                         MOV     R5,R4
11807  042360  005744                         TST     -(R4)           ;S/B SP
11808  042362  104003                  5$:    ERROR   3               ;JSR FAILED
11809  042364  000402                         BR      6$              ;GO EXIT
11810
11811  042366  042344                  7$:    4$                      ;CONTAINS JUMP ADDR
11812  042370  104006                         ERROR   6               ;JSR EXECUTED LIKE A MODE 1 OR 2
11813
11814  042372  010506                  6$:    MOV     R5,SP           ;RESTORE SP JUST IN CASE
11815
11816                                  ;;*****************************************************************
11817                                  ;*TEST 620      JSR MODE 4 TEST
11818                                  ;;*****************************************************************
11819  042374                          TST620:
11820  042374  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11821  042376  012700  000620                 MOV     #620,R0         ;;LOAD R0 WITH TEST NUMBER
11822  042402  013701  042424                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11823  042406  010605                         MOV     SP,R5           ;SAVE THE SP
11824  042410  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11825  042414  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11826  042416  012702  042432                 MOV     #5$,R2          ;DEST ADDR = 4$+2
11827  042422  000257                         CCC                     ;SCOPE SYNC
11828
11829  042424  004442                  2$:    JSR     R4,-(R2)        ;TEST THE JSR - GO TO 4$
11830
11831  042426  104006                  3$:    ERROR   6               ;JSR FAILED TO LOAD THE PC
11832
11833  042430  000401                  4$:    BR      6$              ;JUMPED OK - GO CHECK SP
11834  042432  104005                  5$:    ERROR   5               ;JSR FAILED TO DECREMENT DEST REG
11835
```

```
11836  042434  005726                  6$:     TST     (SP)+           ;RESET SP
11837  042436  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11838  042440  001406                          BEQ     TST621          ;;BR IF YES
11839
11840  042442  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11841  042444  010603                          MOV     SP,R3           ;WAS SP
11842  042446  010504                          MOV     R5,R4
11843  042450  005744                          TST     -(R4)           ;S/B SP
11844  042452  104003                  7$:     ERROR   3               ;JSR FAILED TO PUSH SP
11845
11846  042454  010506                  8$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11847
11848                                  ;;**********************************************************************
11849                                  ;*TEST 621       JSR MODE 5 TEST
11850                                  ;;**********************************************************************
11851  042456                          TST621:
11852  042456  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11853  042460  012700  000621                  MOV     #621,R0         ;;LOAD R0 WITH TEST NUMBER
11854  042464  013701  042506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11855  042470  010605                          MOV     SP,R5           ;SAVE THE SP
11856  042472  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11857  042476  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11858  042500  012702  042536                  MOV     #7$,R2          ;DEST ADDR = [7$ - 2]
11859  042504  000257                          CCC                     ;SCOPE SYNC
11860
11861  042506  004452                  2$:     JSR     R4,@-(R2)       ;TEST THE JSR - GO TO 4$
11862
11863  042510  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11864
11865  042512  005726                  4$:     TST     (SP)+           ;RESET SP
11866  042514  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11867  042516  001411                          BEQ     TST622          ;;BR IF YES
11868
11869  042520  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11870  042522  010603                          MOV     SP,R3           ;WAS SP
11871  042524  010504                          MOV     R5,R4
11872  042526  005744                          TST     -(R4)           ;S/B SP
11873  042530  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11874  042532  000402                          BR      6$              ;GO EXIT
11875
11876  042534  042512                          4$                      ;CONTAINS JUMP ADDRESS
11877  042536  104005                  7$:     ERROR   5               ;JSR EXECUTED LIKE A MODE 1 OR 2
11878
11879  042540  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11880
11881                                  ;;**********************************************************************
11882                                  ;*TEST 622       JSR MODE 6 TEST
11883                                  ;;**********************************************************************
11884  042542                          TST622:
11885  042542  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11886  042544  012700  000622                  MOV     #622,R0         ;;LOAD R0 WITH TEST NUMBER
11887  042550  013701  042572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11888  042554  010605                          MOV     SP,R5           ;SAVE THE SP
11889  042556  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11890  042562  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11891  042564  012702  042576                  MOV     #3$,R2          ;[R2] = BASE DEST ADDR
```

```
11892  042570  000257                  CCC                     ;SCOPE SYNC
11893
11894  042572  004462  000002  2$:     JSR     R4,4$-3$(R2)    ;TEST THE JSR - GO TO 4$
11895
11896  042576  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC OR INDEX FAILED
11897
11898  042600  005726          4$:     TST     (SP)+           ;RESET SP
11899  042602  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11900  042604  001406                  BEQ     TST623          ;;BR IF YES
11901
11902  042606  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11903  042610  010603                  MOV     SP,R3           ;WAS SP
11904  042612  010504                  MOV     R5,R4
11905  042614  005744                  TST     -(R4)           ;S/B SP
11906  042616  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11907  042620  010506                  MOV     R5,SP           ;RESET SP JUST IN CASE
11908
11909                          ;;*******************************************************************
11910                          ;*TEST 623       JSR MODE 7 TEST
11911                          ;;*******************************************************************
11912  042622                  TST623:
11913  042622  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11914  042624  012700  000623          MOV     #623,R0         ;;LOAD R0 WITH TEST NUMBER
11915  042630  013701  042652          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11916  042634  010605                  MOV     SP,R5           ;SAVE THE SP
11917  042636  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11918  042642  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11919  042644  012702  042656          MOV     #3$,R2          ;BASE DEST ADDR = 3$
11920  042650  000257                  CCC                     ;SCOPE SYNC
11921
11922  042652  004472  000024  2$:     JSR     R4,@7$-3$(R2)   ;TEST THE JSR - GO TO 4$ VIA 7$
11923
11924  042656  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11925                                                          ;OR THE INDEX FAILED
11926
11927  042660  005726          4$:     TST     (SP)+           ;RESET SP
11928  042662  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11929  042664  001411                  BEQ     TST624          ;;BR IF YES
11930
11931  042666  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11932  042670  010603                  MOV     SP,R3           ;WAS SP
11933  042672  010504                  MOV     R5,R4
11934  042674  005744                  TST     -(R4)           ;S/B SP
11935  042676  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11936  042700  000402                  BR      6$              ;SKIP TO EXIT
11937
11938  042702  042660          7$:     4$                      ;CONTAINS JUMP ADDR
11939  042704  104005                  ERROR   5               ;JSR WORKED LIKE A MODE 1 OR 2
11940
11941  042706  010506          6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11942
11943                          ;;*******************************************************************
11944                          ;*TEST 624       SOB TEST, [R] = 1, NO BRANCH
11945                          ;;*******************************************************************
11946  042710                  TST624:
11947  042710  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
11948  042712  012700  000624          MOV     #624,R0         ;;LOAD R0 WITH TEST NUMBER
11949  042716  013701  042736          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11950  042722  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
11951  042726  000402                  BR      2$-2            ;GO DO THE SOB
11952
11953  042730  104006          3$:     ERROR   6               ;SOB SHOULDN'T HAVE BRANCHED HERE
11954  042732  000402                  BR      TST625          ;;GO TO SCOPE CALL
11955
11956  042734  000257                  CCC                     ;SYNC INSTR.
11957  042736  077204          2$:     SOB     R2,3$           ;TEST THE SOB
11958
11959                          ;;**********************************************************************
11960                          ;*TEST 625       SOB TEST, [R] = 5, BRANCH 4 TIMES
11961                          ;;**********************************************************************
11962  042740                  TST625:
11963  042740  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11964  042742  012700  000625          MOV     #625,R0         ;;LOAD R0 WITH TEST NUMBER
11965  042746  013701  043000          MOV     @#SOB2,R1       ;GET COPY OF TEST INSTRUCTION WORD
11966  042752  012702  000005          MOV     #5,R2           ;SET SOB COUNTER = 5
11967  042756  012705  177773          MOV     #-5,R5          ;SET UP R5 TO COUNT 5 BRANCHES
11968  042762  000405                  BR      SOB2-2          ;GO DO THE SOB
11969
11970  042764  000474          SOB1:   BR      SOB3            ;USED BY LAST SOB TEST TO TEST MAX OFFSET
11971  042766  000240                  NOP                     ;OFFSET ADJUSTMENT
11972  042770  000240                  NOP
11973
11974  042772  005205          SOB5:   INC     R5              ;COUNT ONE BRANCH
11975  042774  001406                  BEQ     SOBERR          ;BR IF TOO MANY LOOPS BY SOB
11976
11977  042776  000257                  CCC                     ;SCOPE SYNC
11978  043000  077204          SOB2:   SOB     R2,SOB5         ;TEST THE SOB
11979  043002  005702                  TST     R2              ;R2 SHOULD CONTAIN 0
11980  043004  001403                  BEQ     TST626          ;;BR IF IT DOES
11981
11982  043006  104006                  ERROR   6               ;SOB COUNTER NOT ZERO
11983  043010  000401                  BR      TST626          ;;GO TO SCOPE CALL
11984  043012  104006          SOBERR: ERROR   6               ;SOB MADE TOO MANY BRANCHES
11985
11986                          ;;**********************************************************************
11987                          ;*TEST 626       SOB TEST, [R] = 1, FLAGS = 1111
11988                          ;;**********************************************************************
11989  043014                  TST626:
11990  043014  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11991  043016  012700  000626          MOV     #626,R0         ;;LOAD R0 WITH TEST NUMBER
11992  043022  013701  043034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11993  043026  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
11994  043032  000277                  SCC                     ;MAKE N:C = 1111
11995
11996  043034  077202          2$:     SOB     R2,2$-2         ;TEST THE SOB
11997
11998  043036  103003                  BCC     3$              ;BR IF C = 0
11999  043040  102002                  BVC     3$              ;BR IF V = 0
12000  043042  001001                  BNE     3$              ;BR IF Z = 0
12001  043044  100401                  BMI     TST627          ;;BR IF N = 1
12002
12003  043046  104006          3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
```

```
12004
12005                                   ;;*****************************************************************
12006                                   ;*TEST 627       SOB TEST, [R] = 1, FLAGS = 0000
12007                                   ;;*****************************************************************
12008  043050                           TST627:
12009  043050  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12010  043052  012700  000627                  MOV     #627,R0         ;;LOAD R0 WITH TEST NUMBER
12011  043056  013701  043070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12012  043062  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
12013  043066  000257                          CCC                     ;MAKE N:C = 0000
12014
12015  043070  077202                   2$:    SOB     R2,2$-2         ;TEST THE SOB
12016
12017  043072  103403                          BCS     3$              ;BR IF C = 1
12018  043074  102402                          BVS     3$              ;BR IF V = 1
12019  043076  001401                          BEQ     3$              ;BR IF Z = 1
12020  043100  100001                          BPL     TST630          ;;BR IF N = 0
12021
12022  043102  104006                   3$:    ERROR   6               ;SOB ALTERED CODES - SET ONE
12023
12024                                   ;;*****************************************************************
12025                                   ;*TEST 630       SOB TEST, [R] = 5, FLAGS = 1111
12026                                   ;;*****************************************************************
12027  043104                           TST630:
12028  043104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12029  043106  012700  000630                  MOV     #630,R0         ;;LOAD R0 WITH TEST NUMBER
12030  043112  013701  043124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12031  043116  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12032  043122  000277                          SCC                     ;MAKE N:C = 1111
12033
12034  043124  077201                   2$:    SOB     R2,2$           ;TEST THE SOB
12035
12036  043126  103003                          BCC     3$              ;BR IF C = 0
12037  043130  102002                          BVC     3$              ;BR IF V = 0
12038  043132  001001                          BNE     3$              ;BR IF Z = 0
12039  043134  100401                          BMI     TST631          ;;BR IF N = 1
12040
12041  043136  104006                   3$:    ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
12042
12043                                   ;;*****************************************************************
12044                                   ;*TEST 631       SOB TEST, [R] = 5, FLAGS = 0000
12045                                   ;;*****************************************************************
12046  043140                           TST631:
12047  043140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12048  043142  012700  000631                  MOV     #631,R0         ;;LOAD R0 WITH TEST NUMBER
12049  043146  013701  043160                  MOV     @#SOB4,R1       ;GET COPY OF TEST INSTRUCTION WORD
12050  043152  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12051  043156  000257                   SOB3:  CCC                     ;MAKE N:C = 0000
12052
12053  043160  077277                   SOB4:  SOB     R2,SOB1         ;TEST THE SOB
12054
12055  043162  103403                          BCS     3$              ;BR IF C = 1
12056  043164  102402                          BVS     3$              ;BR IF V = 1
12057  043166  001401                          BEQ     3$              ;BR IF Z = 1
12058  043170  100001                          BPL     TST632          ;;BR IF N = 0
12059
```

```
12060  043172  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12061
12062                                  ;;*****************************************************************
12063                                  ;*TEST 632      RTS TEST - N:C = 0000
12064                                  ;;*****************************************************************
12065  043174                          TST632:
12066  043174  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12067  043176  012700  000632                 MOV     #632,R0         ;;LOAD R0 WITH TEST NUMBER
12068  043202  013701  043234                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12069  043206  010605                         MOV     SP,R5           ;SAVE THE SP
12070  043210  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12071  043214  012704  177777          1$:    MOV     #-1,R4          ;R3 SHOULD GET 177777
12072  043220  010506                         MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12073  043222  012703  043242                 MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM [R3]
12074  043226  012746  177777                 MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 WITH 177777
12075  043232  000257                         CCC                     ;N:C = 0000
12076
12077  043234  000203                  2$:    RTS     R3              ;TEST THE RTS - GO TO 4$
12078
12079  043236  104005                  3$:    ERROR   5               ;RTS FAILED TO LOAD THE PC
12080  043240  000415                         BR      10$             ;GO TO EXIT - SCHOOLS OUT
12081
12082  043242  100403                  4$:    BMI     5$              ;N:C = 0000 ?
12083  043244  001402                         BEQ     5$
12084  043246  102401                         BVS     5$
12085  043250  103001                         BCC     6$
12086
12087  043252  104005                  5$:    ERROR   5               ;RTS ALTERED CODES - CLEARED ONE
12088
12089  043254  020403                  6$:    CMP     R4,R3           ;DID R3 GET LOADED FROM STACK ?
12090  043256  001401                         BEQ     8$              ;BR IF YES
12091
12092  043260  104002                  7$:    ERROR   2               ;RTS FAILED TO LOAD REG
12093
12094  043262  020506                  8$:    CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
12095  043264  001404                         BEQ     TST633          ;;BR IF YES
12096
12097  043266  010504                         MOV     R5,R4           ;[R4] = S / B SP
12098  043270  010603                         MOV     SP,R3           ;[R3] = WAS SP
12099  043272  104003                  9$:    ERROR   3               ;RTS FAILED TO POP SP
12100
12101  043274  010506                  10$:   MOV     R5,SP           ;FIX THE SP
12102
12103                                  ;;*****************************************************************
12104                                  ;*TEST 633      RTT TEST - N:C = 1111
12105                                  ;;*****************************************************************
12106  043276                          TST633:
12107  043276  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12108  043300  012700  000633                 MOV     #633,R0         ;;LOAD R0 WITH TEST NUMBER
12109  043304  013701  043346                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12110  043310  012702  177776                 MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12111  043314  010605                         MOV     SP,R5           ;SAVE THE SP
12112  043316  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12113  043322  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12114  043324  012704  000340                 MOV     #340,R4         ;[R4] = S / B PSW AT HTIS POINT
12115  043330  012746  000340                 MOV     #340,-(SP)      ;NEW PSW S / B = 340
```

```
12116  043334  012746  043354          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12117  043340  005037  177776          CLR     @#PSW           ;CLEAR THE PSW
12118  043344  000277                  SCC                     ;N:C = 1111
12119
12120  043346  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12121
12122  043350  104005          3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12123  043352  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12124
12125  043354  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12126  043360  020403                  CMP     R4,R3           ;WAS PSW = 340 ?
12127  043362  001401                  BEQ     6$              ;BR IF IT WAS
12128
12129  043364  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12130
12131  043366  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12132  043370  001404                  BEQ     TST634          ;;BR IF YES
12133
12134  043372  010504                  MOV     R5,R4           ;[R4] = S / B SP
12135  043374  010603                  MOV     SP,R3           ;[R3] = WAS SP
12136  043376  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12137
12138  043400  010506          8$:     MOV     R5,SP           ;FIX THE SP
12139
12140                          ;;*******************************************************************
12141                          ;*TEST 634      RTT TEST - N:C = 0000
12142                          ;;*******************************************************************
12143  043402                  TST634:
12144  043402  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12145  043404  012700  000634          MOV     #634,R0         ;;LOAD R0 WITH TEST NUMBER
12146  043410  013701  043454          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12147  043414  012702  177776          MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12148  043420  010605                  MOV     SP,R5           ;SAVE THE SP
12149  043422  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12150  043426  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12151  043430  012704  000017          MOV     #017,R4         ;[R4] = S / B PSW AT HTIS POINT
12152  043434  012746  000017          MOV     #017,-(SP)      ;NEW PSW S / B = 017
12153  043440  012746  043462          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12154  043444  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW] = 340
12155  043452  000257                  CCC                     ;N:C = 0000
12156
12157  043454  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12158
12159  043456  104005          3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12160  043460  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12161
12162  043462  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12163  043466  020403                  CMP     R4,R3           ;WAS PSW = 017 ?
12164  043470  001401                  BEQ     6$              ;BR IF IT WAS
12165
12166  043472  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12167
12168  043474  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12169  043476  001404                  BEQ     TST635          ;;BR IF YES
12170
12171  043500  010504                  MOV     R5,R4           ;[R4] = S / B SP
```

```
12172  043502  010603                  MOV     SP,R3           ;[R3] = WAS SP
12173  043504  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12174
12175  043506  010506          8$:     MOV     R5,SP           ;FIX THE SP
12176
12177                          ;;**********************************************************************
12178                          ;*TEST 635       MARK INSTRUCTION TEST - N:C=0000
12179                          ;;**********************************************************************
12180  043510                  TST635:
12181  043510  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12182  043512  012700  000635          MOV     #635,R0         ;;LOAD R0 WITH TEST NUMBER
12183  043516  013701  043542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12184  043522  010602                  MOV     SP,R2           ;SAVE SP
12185  043524  012704  125252          MOV     #125252,R4      ;[R5] SHOULD BE 125252
12186  043530  012705  043572          MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12187  043534  010437  043556          MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12188  043540  000257                  CCC                     ;N:C=0000
12189
12190  043542  006405          2$:     MARK+5                  ;TEST THE MARK
12191
12192  043544  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12193  043550  010206                  MOV     R2,SP           ;RESET SP
12194  043552  104005          3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12195
12196  043554  000444                  BR      TST636          ;;GO TO SCOPE EXIT
12197
12198  043556  125252          6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12199
12200  043560  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12201  043564  010206                  MOV     R2,SP           ;RESET SP
12202  043566  104005          5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12203
12204  043570  000436                  BR      TST636          ;;GO TO SCOPE EXIT
12205
12206  043572  100403          4$:     BMI     10$             ;N:C=0000?
12207  043574  001402                  BEQ     10$
12208  043576  102401                  BVS     10$
12209  043600  103011                  BCC     8$
12210
12211  043602  013703  177776  10$:    MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12212  043606  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12213  043612  010206                  MOV     R2,SP           ;RESET SP
12214  043614  012702  177776          MOV     #PSW,R2         ;DEST=PSW
12215  043620  104007          7$:     ERROR   7               ;MARK SET A FLAG
12216  043622  000421                  BR      TST636          ;;GO TO SCOPE EXIT
12217
12218  043624  020627  043560  8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12219  043630  001406                  BEQ     11$             ;BR IF YES
12220  043632  010603                  MOV     SP,R3           ;PUT BAD SP IN R3
12221  043634  012704  043560          MOV     #6$+2,R4        ;S/B SP
12222  043640  010206                  MOV     R2,SP           ;RESET SP
12223  043642  104003          9$:     ERROR   3               ;MARK FAILED TO RESET SP
12224
12225  043644  000410                  BR      TST636          ;;GO TO SCOPE EXIT
12226
12227  043646  020504          11$:    CMP     R5,R4           ;DID MARK RESTORE OLD R5
```

```
12228  043650  001405                    BEQ     12$             ;BR IF YES
12229
12230  043652  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12231  043656  010503                    MOV     R5,R3           ;WAS DEST
12232  043660  010206                    MOV     R2,SP           ;RESET SP
12233  043662  104004                    ERROR   4               ;MARK FAILED TO RESET R5
12234
12235  043664  010206            12$:    MOV     R2,SP           ;RESET SP
12236
12237                            ;;**********************************************************************
12238                            ;*TEST 636      MARK INSTRUCTION TEST - N:C=1111
12239                            ;;**********************************************************************
12240  043666                    TST636:
12241  043666  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12242  043670  012700  000636            MOV     #636,R0         ;;LOAD R0 WITH TEST NUMBER
12243  043674  013701  043720            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12244  043700  010602                    MOV     SP,R2           ;SAVE SP
12245  043702  012704  125252            MOV     #125252,R4      ;[R5] SHOULD BE 125252
12246  043706  012705  043750            MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12247  043712  010437  043734            MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12248  043716  000277                    SCC                     ;N:C=1111
12249
12250  043720  006405            2$:     MARK+5                  ;TEST THE MARK
12251
12252  043722  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12253  043726  010206                    MOV     R2,SP           ;RESET SP
12254  043730  104005            3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12255
12256  043732  000444                    BR      TST637          ;;GO TO SCOPE EXIT
12257
12258  043734  125252            6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12259
12260  043736  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12261  043742  010206                    MOV     R2,SP           ;RESET SP
12262  043744  104005            5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12263
12264  043746  000436                    BR      TST637          ;;GO TO SCOPE EXIT
12265
12266  043750  100003            4$:     BPL     7$              ;N:C=1111
12267  043752  001002                    BNE     7$
12268  043754  102001                    BVC     7$
12269  043756  103411                    BCS     8$
12270
12271  043760  013703  177776    7$:     MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12272  043764  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12273  043770  010206                    MOV     R2,SP           ;RESET SP
12274  043772  012702  177776            MOV     #PSW,R2         ;DEST=PSW
12275  043776  104007                    ERROR   7               ;MARK SET A FLAG
12276  044000  000421                    BR      TST637          ;;GO TO SCOPE EXIT
12277
12278  044002  020627  043736    8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12279  044006  001406                    BEQ     9$              ;BR IF YES
12280  044010  010603                    MOV     SP,R3           ;PUT BAD SP IN R3
12281  044012  012704  043736            MOV     #6$+2,R4        ;S/B SP
12282  044016  010206                    MOV     R2,SP           ;RESET SP
12283  044020  104003                    ERROR   3               ;MARK FAILED TO RESET SP
```

```
12284
12285  044022  000410                    BR      TST637          ;;GO TO SCOPE EXIT
12286
12287  044024  020504            9$:     CMP     R5,R4           ;DID MARK RESTORE OLD R5
12288  044026  001405                    BEQ     10$             ;BR IF YES
12289
12290  044030  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12291  044034  010503                    MOV     R5,R3           ;WAS DEST
12292  044036  010206                    MOV     R2,SP           ;RESET SP
12293  044040  104004                    ERROR   4               ;MARK FAILED TO RESET R5
12294
12295  044042  010206            10$:    MOV     R2,SP           ;RESET SP
12296
12297                            ;;**********************************************************************
12298                            ;*TEST 637      BASIC LINE CLOCK RESPONSE TEST
12299                            ;;**********************************************************************
12300  044044                    TST637:
12301  044044  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12302  044046  012700  000637            MOV     #637,R0         ;;LOAD R0 WITH TEST NUMBER
12303  044052  013701  044102            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12304  044056  010605                    MOV     SP,R5           ;SAVE SP
12305  044060  012702  177546            MOV     #LKCSR,R2       ;[R2] = LINE CLOCK ADDRESS
12306  044064  010737  001010            MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12307  044070  010506            1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12308  044072  012737  044106  000004    MOV     #4$,@#4         ;GO TO 4$ IF BUS TIMEOUT
12309  044100  000257                    CCC                     ;SCOPE SYNC
12310
12311  044102  005712            2$:     TST     (R2)            ;REFERENCE LKCSR ADDR
12312
12313  044104  000404                    BR      6$              ;GO TO EXIT
12314
12315  044106  012737  061220  000004    MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12316  044114  104006            3$:     ERROR   6               ;LKCSR FAILED TO RESPOND
12317
12318  044116  010506            6$:     MOV     R5,SP           ;RESET SP
12319  044120  012737  061220  000004    MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12320
12321                            ;;**********************************************************************
12322                            ;*TEST 640      LINE CLOCK TEST - LKCSR BIT 7 SET
12323                            ;;**********************************************************************
12324  044126                    TST640:
12325  044126  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12326  044130  012700  000640            MOV     #640,R0         ;;LOAD R0 WITH TEST NUMBER
12327  044134  013701  044152            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12328  044140  012702  177546            MOV     #LKCSR,R2       ;DEST ADDR = 177546
12329  044144  012704  000200            MOV     #200,R4         ;[LKCSR] S / B = 200
12330  044150  000257                    CCC                     ;SCOPE SYNC
12331
12332  044152  030412            2$:     BIT     R4,(R2)         ;TEST BIT 7 IN LKCSR
12333
12334  044154  001002                    BNE     TST641          ;;BR IF IT'S SET
12335
12336  044156  011203                    MOV     (R2),R3         ;GET WAS DATA
12337  044160  104001            3$:     ERROR   1               ;BIT 7 NOT SET IN LKCSR
12338
12339                            ;;**********************************************************************
```

```
12340                                   ;*TEST 641      LINE CLOCK TEST - LKCSR BIT 6 CLEAR
12341                                   ;;**************************************************************
12342  044162                           TST641:
12343  044162  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12344  044164  012700  000641                   MOV     #641,R0         ;;LOAD R0 WITH TEST NUMBER
12345  044170  013701  044206                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12346  044174  012702  177546                   MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12347  044200  012704  000200                   MOV     #200,R4         ;[LKCSR] S / B = 200
12348  044204  000257                           CCC                     ;SCOPE SYNC
12349
12350  044206  032712  000100           2$:     BIT     #100,(R2)       ;TEST BIT 6 IN LKCSR
12351
12352  044212  001402                           BEQ     TST642          ;;BR IF CLEAR
12353
12354  044214  011203                           MOV     (R2),R3         ;GET WAS DATA
12355  044216  104001                   3$:     ERROR   1               ;BIT 6 (INTR. ENAB.) IN LKCSR WAS SET
12356
12357                                   ;;**************************************************************
12358                                   ;*TEST 642      LINE CLOCK TEST - LKCSR BIT 6 SET
12359                                   ;;**************************************************************
12360  044220                           TST642:
12361  044220  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12362  044222  012700  000642                   MOV     #642,R0         ;;LOAD R0 WITH TEST NUMBER
12363  044226  013701  044276                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12364  044232  010605                           MOV     SP,R5           ;SAVE SP
12365  044234  012702  177546                   MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12366  044240  012704  000300                   MOV     #300,R4         ;[LKCSR] S / B = 300
12367  044244  010737  001010                   MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12368  044250  012737  044312  000100   1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR IN CASE LOGIC
12369  044256  012737  000340  000102           MOV     #340,@#102      ;FAULT CAUSES ATL INTERRUPT
12370  044264  010506                           MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12371  044266  012737  000340  177776           MOV     #340,@#PSW      ;SET PRIORITY TO LEVEL 7
12372  044274  000257                           CCC                     ;SCOPE SYNC
12373
12374  044276  052712  000100           2$:     BIS     #100,(R2)       ;SET BIT 6 IN LKCSR
12375
12376  044302  020412                           CMP     R4,(R2)         ;RESULT CORRECT?
12377  044304  001402                           BEQ     4$              ;BR IF YES
12378
12379  044306  011203                           MOV     (R2),R3         ;GET WAS DATA
12380  044310  104001                   3$:     ERROR   1               ;BIT 6 FAILED TO SET IN LKCSR
12381
12382  044312  042737  000102  000100   4$:     BIC     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLOCK VECTOR
12383  044320  005037  000102                   CLR     @#102
12384  044324  042712  000100                   BIC     #100,(R2)       ;TURN OF LINE CLK INTR. ENAB.
12385  044330  010506                           MOV     R5,SP           ;RESET SP
12386
12387                                   ;;**************************************************************
12388                                   ;*TEST 643      LINE CLK BASIC INTERRUPT TEST
12389                                   ;;**************************************************************
12390  044332                           TST643:
12391  044332  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12392  044334  012700  000643                   MOV     #643,R0         ;;LOAD R0 WITH TEST NUMBER
12393  044340  013701  044406                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12394  044344  010605                           MOV     SP,R5           ;SAVE SP
12395  044346  012702  177546                   MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
```

```
044352  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
044356  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
044360  005004                          CLR     R4              ;INITIALIZE TIMER
044362  012737  044424  000100          MOV     #4$,@#100       ;SET UP LINE CLOCK VECTOR TO TO
044370  012737  000340  000102          MOV     #340,@#102      ;TO 4$ WITH PROCESSOR PRIORITY = 7
044376  005012                          CLR     (R2)            ;CLEAR LKCSR
044400  005037  177776                  CLR     @#PSW           ;SET PRIORITY TO LEVEL 000
044404  000257                          CCC                     ;SCOPE SYNC

044406  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPT

044412  005304                          DEC     R4              ;WAIT FOR INTR - REPORT ERROR IF
044414  001376                          BNE     .-2             ;R4 GOES TO 000000

044416  042712  000100                  BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
044422  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTERRUPT

044424  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
044430  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLK VECTOR
044436  005037  000102                  CLR     @#102
044442  010506                          MOV     R5,SP           ;RESET SP
044444  005037  177776                  CLR     @#PSW           ;RESET PRIORITY TO LEVEL 0

                                ;;*******************************************************************
                                ;*TEST 644       RESET TEST - N:C = 1111
                                ;;*******************************************************************
044450                          TST644:
044450  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
044452  012700  000644                  MOV     #644,R0         ;;LOAD R0 WITH TEST NUMBER
044456  013701  044510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
044462  012737  000001  001110          MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
044470  012702  177564                  MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
044474  012737  000340  177776          MOV     #340,@#PSW      ;MAKE PRTY. BITS ALL 1'S
044502  052712  000004                  BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
044506  000277                          SCC                     ;N:C = 1111

044510  000005                  2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT

044512  013705  177776                  MOV     @#PSW,R5        ;SAVE THE PSW
044516  032712  000004                  BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
044522  001403                          BEQ     4$              ;BR IF YES

044524  042712  000004                  BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
044530  104006                  3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT

044532  022705  000357          4$:     CMP     #357,R5         ;DID RESET ALTER THE PSW ??
044536  001406                          BEQ     6$              ;BR IF NOT

044540  012704  000357                  MOV     #357,R4         ;[R4] = S/B PSW
044544  010503                          MOV     R5,R3           ;[R3] = WAS PSW
044546  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
044552  104001                  5$:     ERROR   1               ;RESET ALTERED THE PSW

044554  005037  177776          6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
044560  042737  000004  177564          BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
```

```
                                        ;;*******************************************************************
                                        ;*TEST 645      RESET TEST - N:C = 0000
                                        ;;*******************************************************************
044566                                  TST645:
044566  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
044570  012700  000645                          MOV     #645,R0         ;;LOAD R0 WITH TEST NUMBER
044574  013701  044624                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
044600  012737  000001  001110                  MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
044606  012702  177564                          MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
044612  005037  177776                          CLR     @#PSW           ;MAKE PRTY. BITS ALL 0'S
044616  052712  000004                          BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
044622  000257                                  CCC                     ;N:C = 0000

044624  000005                          2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT

044626  013705  177776                          MOV     @#PSW,R5        ;SAVE THE PSW
044632  032712  000004                          BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
044636  001403                                  BEQ     4$              ;BR IF YES

044640  042712  000004                          BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
044644  104006                          3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT

044646  022705  000000                  4$:     CMP     #0,R5           ;DID RESET ALTER THE PSW ??
044652  001406                                  BEQ     6$              ;BR IF NOT

044654  012704  000357                          MOV     #357,R4         ;[R4] = S/B PSW
044660  010503                                  MOV     R5,R3           ;[R3] = WAS PSW
044662  012702  177776                          MOV     #PSW,R2         ;DEST = PSW
044666  104001                          5$:     ERROR   1               ;RESET ALTERED THE PSW

044670  005037  177776                  6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
044674  042737  000004  177564                  BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF

                                        ;;*******************************************************************
                                        ;*TEST 646      WAIT INSTRUCTION TEST - [PSW] = 151
                                        ;;*******************************************************************
044702                                  TST646:
044702  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
044704  012700  000646                          MOV     #646,R0         ;;LOAD R0 WITH TEST NUMBER
044710  013701  044776                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
044714  010605                                  MOV     SP,R5           ;SAVE THE SP
044716  010737  001010                          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
044722  012702  177564                  1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
044726  012737  045014  000064                  MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
044734  012737  000200  000066                  MOV     #200,@#66       ;AT LEVEL 4
044742  010506                                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
044744  005012                                  CLR     (R2)            ;INIT DL11 XCSR
044746  005003                                  CLR     R3              ;INIT TIMER

044750  105712                          3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
044752  100403                                  BMI     5$              ;BR IF YES
044754  005303                                  DEC     R3              ;COUNT THE TIMER
044756  001374                                  BNE     3$              ;BR IF NO TIMEOUT
044760  000440                                  BR      9$              ;GO REPORT TIMEOUT

044762  012737  000140  177776          5$:     MOV     #140,@#PSW      ;SET PSW PRTY BITS TO LEVEL 3
```

```
12508  044770  000277                      SCC             ;N:C=1111
12509  044772  152712  000100              BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1001
12510
12511  044776  000001              2$:     WAIT            ;TEST THE WAIT-GO TO 4$ ON INTR
12512
12513  045000  012737  000340  177776      MOV     #340,@#PSW      ;LOCK OUT INTR
12514  045006  005012                      CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12515  045010  104006                      ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12516  045012  000424                      BR      8$              ;GO EXIT THIS TEST
12517
12518  045014  042712  000100      4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12519  045020  022716  045000              CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12520  045024  001402                      BEQ     6$              ;BR IF YES
12521
12522  045026  104006                      ERROR   6               ;WAIT NOT FETCHED PROPERLY
12523  045030  000415                      BR      8$              ;GO EXIT THE TEST
12524
12525  045032  022766  000151  000002  6$: CMP     #151,2(SP)      ;DID 'WAIT'' ALTER THE PSW ??
12526  045040  001411                      BEQ     8$              ;BR IF YES
12527
12528  045042  012704  000151              MOV     #151,R4         ;[R4] = S/B PSW
12529  045046  016603  000002              MOV     2(SP),R3        ;[R3] = WAS PSW
12530  045052  012702  177776              MOV     #PSW,R2         ;DEST = PSW
12531  045056  104001              7$:     ERROR   1               ;'WAIT'' ALTERED THE PSW
12532  045060  000401                      BR      8$              ;GOT TO EXIT TEST
12533
12534  045062  104006              9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12535
12536  045064  010506              8$:     MOV     R5,SP           ;RESET THE SP
12537  045066  005037  177776              CLR     @#PSW           ;CLEAR OUT THE PSW
12538  045072  005012                      CLR     (R2)            ;TURN OFF DL11 INTR.
12539  045074  012737  000066  000064      MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12540  045102  005037  000066              CLR     @#66
12541
12542                          ;;*********************************************************************
12543                          ;*TEST 647      WAIT INSTRUCTION TEST - [PSW] = 010
12544                          ;;*********************************************************************
12545  045106                  TST647:
12546  045106  000004                      SCOPE           ;CALL THE SCOPE LOOP UTILITY
12547  045110  012700  000647              MOV     #647,R0         ;;LOAD R0 WITH TEST NUMBER
12548  045114  013701  045200              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12549  045120  010605                      MOV     SP,R5           ;SAVE THE SP
12550  045122  010737  001010              MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12551  045126  012702  177564      1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12552  045132  012737  045216  000064      MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12553  045140  012737  000200  000066      MOV     #200,@#66       ;AT LEVEL 4
12554  045146  010506                      MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12555  045150  005012                      CLR     (R2)            ;INIT DL11 XCSR
12556  045152  005003                      CLR     R3              ;INIT TIMER
12557
12558  045154  105712              3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12559  045156  100403                      BMI     5$              ;BR IF YES
12560  045160  005303                      DEC     R3              ;COUNT THE TIMER
12561  045162  001374                      BNE     3$              ;BR IF NO TIMEOUT
12562  045164  000437                      BR      9$              ;GO REPORT TIMEOUT
12563
```

```
12564  045166  005037  177776          5$:     CLR     @#PSW           ;SET PSW PRTY BITS TO LEVEL 0
12565  045172  000257                          CCC                     ;N:C=0000
12566  045174  152712  000100                  BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1000
12567
12568  045200  000001                  2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12569
12570  045202  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
12571  045210  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12572  045212  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12573  045214  000424                          BR      8$              ;GO EXIT THIS TEST
12574
12575  045216  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12576  045222  022716  045202                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12577  045226  001402                          BEQ     6$              ;BR IF YES
12578
12579  045230  104006                          ERROR   6               ;WAIT NOT FETCHED PROPERLY
12580  045232  000415                          BR      8$              ;GO EXIT THE TEST
12581
12582  045234  022766  000010  000002  6$:     CMP     #010,2(SP)      ;DID 'WAIT' ALTER THE PSW ??
12583  045242  001411                          BEQ     8$              ;BR IF NO
12584
12585  045244  012704  000010                  MOV     #010,R4         ;[R4] = S/B PSW
12586  045250  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
12587  045254  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12588  045260  104001                  7$:     ERROR   1               ;'WAIT' ALTERED THE PSW
12589  045262  000401                          BR      8$              ;GOT TO EXIT TEST
12590
12591  045264  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12592
12593  045266  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12594  045270  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
12595  045274  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
12596  045276  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12597  045304  005037  000066                  CLR     @#66
12598
12599                                  ;;*******************************************************************
12600                                  ;*TEST 650      BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
12601                                  ;;*******************************************************************
12602  045310                          TST650:
12603  045310  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12604  045312  012700  000650                  MOV     #650,R0         ;;LOAD R0 WITH TEST NUMBER
12605  045316  013701  045364                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12606  045322  010605                          MOV     SP,R5           ;SAVE THE SP
12607  045324  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12608  045330  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12609  045334  012737  045402  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12610  045342  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12611  045350  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12612  045352  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12613  045354  012737  000040  177776          MOV     #40,@#PSW       ;SET CPU PRIORITY TO LEVEL 1
12614  045362  000257                          CCC                     ;SCOPE SYNC
12615
12616  045364  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12617
12618  045370  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12619  045372  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
```

```
12620
12621  045374  042712  000100          BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12622  045400  104006          3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 1
12623
12624  045402  042712  000100    4$:   BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
```

```
12625  045406  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12626  045414  005037  000102                  CLR     @#102
12627  045420  010506                          MOV     R5,SP           ;RESET THE SP
12628  045422  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12629
12630                                  ;;*******************************************************************
12631                                  ;*TEST 651       BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
12632                                  ;;*******************************************************************
12633  045426                          TST651:
12634  045426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12635  045430  012700  000651                  MOV     #651,R0         ;;LOAD R0 WITH TEST NUMBER
12636  045434  013701  045502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12637  045440  010605                          MOV     SP,R5           ;SAVE THE SP
12638  045442  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12639  045446  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12640  045452  012737  045520  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12641  045460  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12642  045466  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12643  045470  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12644  045472  012737  000100  177776          MOV     #100,@#PSW      ;SET CPU PRIORITY TO LEVEL 2
12645  045500  000257                          CCC                     ;SCOPE SYNC
12646
12647  045502  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12648
12649  045506  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12650  045510  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12651
12652  045512  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12653  045516  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 2
12654
12655  045520  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12656  045524  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12657  045532  005037  000102                  CLR     @#102
12658  045536  010506                          MOV     R5,SP           ;RESET THE SP
12659  045540  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12660
12661                                  ;;*******************************************************************
12662                                  ;*TEST 652       BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
12663                                  ;;*******************************************************************
12664  045544                          TST652:
12665  045544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12666  045546  012700  000652                  MOV     #652,R0         ;;LOAD R0 WITH TEST NUMBER
12667  045552  013701  045620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12668  045556  010605                          MOV     SP,R5           ;SAVE THE SP
12669  045560  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12670  045564  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12671  045570  012737  045636  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12672  045576  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12673  045604  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12674  045606  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12675  045610  012737  000140  177776          MOV     #140,@#PSW      ;SET CPU PRIORITY TO LEVEL 3
12676  045616  000257                          CCC                     ;SCOPE SYNC
12677
12678  045620  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12679
12680  045624  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
```

```
12681  045626  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12682
12683  045630  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12684  045634  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 3
12685
12686  045636  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12687  045642  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12688  045650  005037  000102                  CLR     @#102
12689  045654  010506                          MOV     R5,SP           ;RESET THE SP
12690  045656  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12691
12692                                  ;;**********************************************************************
12693                                  ;*TEST 653       BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
12694                                  ;;**********************************************************************
12695  045662                          TST653:
12696  045662  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12697  045664  012700  000653                  MOV     #653,R0         ;;LOAD R0 WITH TEST NUMBER
12698  045670  013701  045736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12699  045674  010605                          MOV     SP,R5           ;SAVE THE SP
12700  045676  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12701  045702  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12702  045706  012737  045754  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12703  045714  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12704  045722  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12705  045724  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12706  045726  012737  000200  177776          MOV     #200,@#PSW      ;SET CPU PRIORITY TO LEVEL 4
12707  045734  000257                          CCC                     ;SCOPE SYNC
12708
12709  045736  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12710
12711  045742  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12712  045744  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12713
12714  045746  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12715  045752  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 4
12716
12717  045754  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12718  045760  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12719  045766  005037  000102                  CLR     @#102
12720  045772  010506                          MOV     R5,SP           ;RESET THE SP
12721  045774  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12722
12723                                  ;;**********************************************************************
12724                                  ;*TEST 654       BR PRIORITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12725                                  ;;**********************************************************************
12726  046000                          TST654:
12727  046000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12728  046002  012700  000654                  MOV     #654,R0         ;;LOAD R0 WITH TEST NUMBER
12729  046006  013701  046054                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12730  046012  010605                          MOV     SP,R5           ;SAVE THE SP
12731  046014  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12732  046020  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12733  046024  012737  046072  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12734  046032  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12735  046040  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12736  046042  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
```

```
12737 046044 012737 000240 177776          MOV     #240,@#PSW      ;SET CPU PRIORITY TO LEVEL 5
12738 046052 000257                        CCC                     ;SCOPE SYNC
12739
12740 046054 052712 000100          2$:    BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12741
12742 046060 005304                        DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12743 046062 001376                        BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12744
12745 046064 042712 000100                 BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12746 046070 104006                 3$:    ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 5
12747
12748 046072 042712 000100          4$:    BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12749 046076 012737 000102 000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12750 046104 005037 000102                 CLR     @#102
12751 046110 010506                        MOV     R5,SP           ;RESET THE SP
12752 046112 005037 177776                 CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12753
12754                               ;;*****************************************************************
12755                               ;*TEST 655      BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12756                               ;;*****************************************************************
12757 046116                       TST655:
12758 046116 000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12759 046120 012700 000655                 MOV     #655,R0         ;;LOAD R0 WITH TEST NUMBER
12760 046124 013701 046204                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12761                               .SBTTL USER CONTROLLED BREAKPOINT -- BIT13
12762 046130 032737 020000 063234          BIT     #BIT13,@#BPTLOC ;BREAKPOINT HALT SET ??
12763 046136 001401                        BEQ     .+4             ;BR IF NOT
12764 046140 000000                        HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
12765 046142 010605                        MOV     SP,R5           ;SAVE THE SP
12766 046144 010737 001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12767 046150 012702 177546          1$:    MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12768 046154 012737 046216 000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12769 046162 012737 000340 000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12770 046170 010506                        MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12771 046172 005004                        CLR     R4              ;INITIALIZE R4 AS TIMER
12772 046174 012737 000300 177776          MOV     #300,@#PSW      ;SET CPU PRIORITY TO LEVEL 6
12773 046202 000257                        CCC                     ;SCOPE SYNC
12774
12775 046204 052712 000100          2$:    BIS     #100,(R2)       ;ENABLE INTERRUPTS
12776
12777 046210 005304                        DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12778 046212 001376                        BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12779 046214 000403                        BR      6$              ;GO TO EXIT - ALL OK
12780
12781 046216 042712 000100          4$:    BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12782 046222 104006                 3$:    ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 6
12783
12784 046224 042712 000100          6$:    BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12785 046230 012737 000102 000100          MOV     #102,@#100      ;RESET THE TRAP CATCHER IN THE VECTOR
12786 046236 005037 000102                 CLR     @#102
12787 046242 010506                        MOV     R5,SP           ;RESET SP JUST IN CASE
12788 046244 005037 177776                 CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12789
12790                               ;;*****************************************************************
12791                               ;*TEST 656      BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12792                               ;;*****************************************************************
```

```
12793  046250                          TST656:
12794  046250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12795  046252  012700  000656                  MOV     #656,R0         ;;LOAD R0 WITH TEST NUMBER
12796  046256  013701  046324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12797  046262  010605                          MOV     SP,R5           ;SAVE THE SP
12798  046264  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12799  046270  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12800  046274  012737  046336  000064          MOV     #4$,@#64        ;IF INTR OCCURS - GO TO 4$
12801  046302  012737  000340  000066          MOV     #340,@#66       ;WITH CPU PRIORITY AT LEVEL 7
12802  046310  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12803  046312  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12804  046314  012737  000340  177776          MOV     #340,@#PSW      ;SET CPU PRIORITY TO LEVEL 7
12805  046322  000257                          CCC                     ;SCOPE SYNC
12806
12807  046324  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12808
12809  046330  005304                          DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12810  046332  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12811  046334  000403                          BR      6$              ;GO TO EXIT - ALL OK
12812
12813  046336  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12814  046342  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 7
12815
12816  046344  042712  000100          6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12817  046350  012737  000066  000064          MOV     #66,@#64        ;RESET THE TRAP CATCHER IN THE VECTOR
12818  046356  005037  000066                  CLR     @#66
12819  046362  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12820  046364  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12821
12822                                  ;;**********************************************************************
12823                                  ;*TEST 657      ''CLR  @#PSW'' ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12824                                  ;THIS TEST VERIFIES THAT IF A ''BR'' REQUEST IS PENDING WHEN A ''CLR  @#PSW''
12825                                  ;IS EXECUTED TO LOWER THE CPU PRIORITY, THE REQUEST IS GRANTED BEFORE
12826                                  ;EXECUTION OF THE INSTRUCTION FOLLOWING THE ''CLR''
12827                                  ;;**********************************************************************
12828  046370                          TST657:
12829  046370  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12830  046372  012700  000657                  MOV     #657,R0         ;;LOAD R0 WITH TEST NUMBER
12831  046376  013701  046470                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12832  046402  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12833  046406  010605                          MOV     SP,R5           ;SAVE THE SP
12834  046410  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12835  046414  012737  046476  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR TO GO TO 4$
12836  046422  012737  000300  000102          MOV     #300,@#102
12837  046430  010506                          MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12838  046432  005004                          CLR     R4              ;INITIALIZE TIMER FO KW
12839  046434  005003                          CLR     R3              ;CLEAR SOFTWARE FLAG
12840  046436  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12841  046444  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12842  046450  042712  000200                  BIC     #200,(R2)       ;CLEAR LINE CLOCK READY
12843  046454  105712                  11$:    TSTB    (R2)            ;LCLK READY TO INTR ??
12844  046456  100403                          BMI     12$             ;BR IF YES
12845  046460  005304                          DEC     R4              ;COUNT THE TIMER
12846  046462  001374                          BNE     11$             ;BR IF NO TIMEOUT
12847  046464  000411                          BR      6$              ;GO REPORT TIMEOUT
12848  046466  000257                  12$:    CCC                     ;SCOPE SYNC
```

```
12849
12850  046470  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - LCLK SHOULD INTERRUPT
12851                                                                   ;BEFORE FETCHING NEXT INSTRUCTION
12852  046474  005103                          COM     R3              ;SHOULD NOT BE FETCHED
12853  046476  005012                  4$:     CLR     (R2)            ;DISABLE THE LCLK INTR
12854  046500  005703                          TST     R3              ;DID SOFTWARE FLAG GET SET ??
12855  046502  001404                          BEQ     8$              ;BR IF NOT - IT WORKED OK
12856  046504  104006                  3$:     ERROR   6               ;LCLK FAILED TO INTR ONTIME
12857  046506  000402                          BR      8$              ;GO EXIT
12858
12859  046510  005012                  6$:     CLR     (R2)            ;DISABLE  LCLK INTR
12860  046512  104006                  5$:     ERROR   6               ;LINE CLK TIMED OUT
12861
12862  046514  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12863  046516  012737  000102  000100          MOV     #102,@#100      ;RESTORE THE LINE CLK TRAPCATCHER
12864  046524  005037  000102                  CLR     @#102
12865
12866                                  ;;**********************************************************************
12867                                  ;*TEST 660      'BR6 VS BR4'' PRIORITY ARBITRATION TEST
12868                                  ;THIS TEST VERIFIES THAT IF BOTH A 'BR4'' AND A 'BR6'' REQUEST ARE
12869                                  ;PENDING WHEN THE CPU PRIORITY IS LOWERED TO ALLOW INTRS. THAT 'BR6''
12870                                  ;REQUEST IS GRANTED FIRST EVEN THOUGH THE 'BR4'' REQUEST MAY HAVE
12871                                  ;OCCURRED FIRST
12872                                  ;;**********************************************************************
12873  046530                          TST660:
12874  046530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12875  046532  012700  000660                  MOV     #660,R0         ;;LOAD R0 WITH TEST NUMBER
12876  046536  013701  046676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12877  046542  010605                          MOV     SP,R5           ;SAVE THE SP
12878  046544  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12879  046550  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12880  046554  012703  177564                  MOV     #XCSR,R3        ;R3 POINTS TO DL11 XCSR
12881  046560  012737  046706  000100          MOV     #4$,@#100       ;SET UP THE LCLK VECTOR - GO TO 4$
12882  046566  012737  000300  000102          MOV     #300,@#102
12883  046574  012737  046740  000064          MOV     #8$,@#64        ;SET UP THE DL11 VECTOR - GO TO 8$
12884  046602  012737  000200  000066          MOV     #200,@#66
12885  046610  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12886  046612  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12887  046620  005037  063312                  CLR     @#MBUF0         ;INIT TIMER
12888  046624  005037  063316                  CLR     @#MBUF1         ;CLEAR DL11 INTR FLAG
12889  046630  005004                          CLR     R4              ;INIT TIMER
12890  046632  052713  000100                  BIS     #100,(R3)       ;ENABLE DL11 XMIT INTR
12891  046636  105713                  11$:    TSTB    (R3)            ;XMIT READY SET ??
12892  046640  100403                          BMI     12$             ;BR IF YES
12893  046642  005304                          DEC     R4              ;COUNT THE TIMER
12894  046644  001374                          BNE     11$             ;BR IF NO TIMEOUT
12895  046646  000443                          BR      5$              ;GO REPORT TIMEOUT FOR DL11
12896
12897  046650  005004                  12$:    CLR     R4              ;INIT THE TIMER AGAIN
12898  046652  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12899  046656  042712  000200                  BIC     #200,(R2)       ;CLEAR THE LINE CLOCK READY BIT
12900  046662  105712                  13$:    TSTB    (R2)            ;LCLK READY TO INTR
12901  046664  100403                          BMI     14$             ;BR IF YES
12902  046666  005304                          DEC     R4              ;COUNT THE TIMER
12903  046670  001374                          BNE     13$             ;BR IF NO TIMEOUT
12904  046672  000436                          BR      7$              ;GO REPORT LINE CLK TIMEOUT
```

```
12905  046674  000257                14$:    CCC                     ;SCOPE SYNC
12906
12907  046676  005037  177776        2$:     CLR     @#PSW           ;ALLOW INTRS - KW SHOULD INTR FIRST
12908
12909  046702  005137  063312                COM     @#MBUF0         ;SET SOFTWARE FLAG IF FETCHED
12910  046706  005013                4$:     CLR     (R3)            ;DISABLE BOTH INTERRUPTS
12911  046710  005012                        CLR     (R2)
12912  046712  005737  063312                TST     @#MBUF0         ;DID SOFTWARE FLAG GET SET ??
12913  046716  001402                        BEQ     6$              ;BR IF NOT
12914
12915  046720  104006                3$:     ERROR   6               ;LINE CLK INTR OCCURRED TOO LATE
12916  046722  000425                        BR      9$              ;GO TO EXIT
12917
12918  046724  005737  063316        6$:     TST     @#MBUF1         ;DID DL11 SOFTWARE FLAG SET ??
12919  046730  001422                        BEQ     9$              ;BR IF NOT
12920
12921  046732  010302                        MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12922  046734  104006                        ERROR   6               ;DL11 INTERRUPTED THE KW11
12923  046736  000417                        BR      9$              ;GO TO EXIT TEST
12924
12925  046740  005137  063316        8$:     COM     @#MBUF1         ;FLAG THE DL11 INTR
12926  046744  005013                        CLR     (R3)            ;DISABLE BOTH INTR ENABLES
12927  046746  005012                        CLR     (R2)
12928  046750  010302                        MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12929  046752  104006                        ERROR   6               ;DL11 SHOULD NOT HAVE INTERRUPTED
12930  046754  000410                        BR      9$              ;GO EXIT TEST
12931
12932  046756  005012                5$:     CLR     (R2)            ;DISABLE THE INTR ENABLES
12933  046760  005013                        CLR     (R3)
12934  046762  010302                        MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12935  046764  104006                        ERROR   6               ;DL11 TIMEOUT
12936  046766  000403                        BR      9$              ;GO TO EXIT
12937
12938  046770  005012                7$:     CLR     (R2)            ;DISABLE INTR ENABLES
12939  046772  005013                        CLR     (R3)
12940  046774  104006                        ERROR   6               ;KW11 TIMEOUT
12941
12942  046776  010506                9$:     MOV     R5,SP           ;RESET THE SP
12943  047000  005037  177776                CLR     @#PSW           ;RESET THE CPU PRIORITY
12944  047004  012737  000102  000100        MOV     #102,@#100      ;RESTORE LCLK VECTOR
12945  047012  005037  000102                CLR     @#102
12946  047016  012737  000066  000064        MOV     #66,@#64        ;RESTORE THE DL11 XMIT VECTOR
12947  047024  005037  000066                CLR     @#66
12948
12949                                ;*****************************************************************
12950                                ;/////////////COMBINED INSTRUCTION EXERCISER SECTION /////////////
12951                                ;*****************************************************************
12952
12953                                ;;*************************************************************
12954                                ;*TEST 661       'BPT'' TRAP ! INKAGE TEST
12955                                ;;*************************************************************
12956  047030                        TST661:
12957  047030  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12958  047032  012700  000661                MOV     #661,R0         ;;LOAD R0 WITH TEST NUMBER
12959  047036  013701  047062                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12960  047042  010605                        MOV     SP,R5           ;SAVE THE SP
```

```
12961  047044  010737  001010              MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12962  047050  012737  047066  000014  1$: MOV     #4$,@#14        ;GO TO 4$ ON 'BPT'' TRAP
12963  047056  010506                      MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12964  047060  000257                      CCC                     ;SCOPE SYNC
12965
12966  047062  000003                  2$: BPT                     ;TEST THE 'BPT'' - GO TO 4$
12967
12968  047064  104005                  3$: ERROR   5               ;BPT FAILED TO TRAP
12969
12970  047066  010506                  4$: MOV     R5,SP           ;RESET THE SP
12971  047070  012737  000016  000014      MOV     #16,@#14        ;RESTORE THE VECTOR
12972
12973                                  ;;*****************************************************************
12974                                  ;*TEST 662      RED ZONE OVERFLOW TEST - MOV R,-(SP)
12975                                  ;;*****************************************************************
12976  047076                          TST662:
12977  047076  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12978  047100  012700  000662              MOV     #662,R0         ;;LOAD R0 WITH TEST NUMBER
12979  047104  013701  047144              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12980  047110  010605                      MOV     SP,R5           ;SAVE SP
12981  047112  013704  000004              MOV     @#4,R4          ;SAVE T.O. VECTOR
12982  047116  013703  000336              MOV     @#336,R3        ;SAVE VECTOR AT 336
12983  047122  012737  047162  000004      MOV     #4$,@#4 ;GO TO 4$ ON OVFLW
12984  047130  012737  125252  000336      MOV     #125252,@#336   ;INIT. [336]
12985  047136  012706  000340              MOV     #340,SP         ;SET SP TO CAUSE RED ZONE TRAP
12986  047142  000257                      CCC                     ;SCOPE SYNC
12987
12988  047144  010046                  2$: MOV     R0,-(SP)        ;FORCE RED ZONE TRAP - GO TO 4$
12989
12990  047146  010437  000004              MOV     R4,@#4          ;RESTORE T.O. VECTOR
12991  047152  010637  001074              MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12992  047156  010506                      MOV     R5,SP           ;RESET SP FOR ERROR CALL
12993  047160  104005                  3$: ERROR   5               ;MOV FAILED TO CAUSE TRAP
12994
12995  047162  010437  000004          4$: MOV     R4,@#4          ;RESTORE T.O. VECTOR
12996  047166  022706  000000              CMP     #0,SP           ;[SP]=0?
12997  047172  001404                      BEQ     6$              ;BE IF YES
12998
12999  047174  010637  001074              MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13000  047200  010506                      MOV     R5,SP           ;RESET SP FOR ERROR CALL
13001  047202  104005                  5$: ERROR   5               ;SP NOT BEING JAMMED TO 4
13002
13003  047204  022737  125252  000336  6$: CMP     #125252,@#336   ;DID PUSH OCCUR IN YELLOW ZONE?
13004  047212  001404                      BEQ     8$              ;BR IF NOT
13005
13006  047214  010637  001074              MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13007  047220  010506                      MOV     R5,SP           ;RESET SP FOR ERROR CALL
13008  047222  104005                  7$: ERROR   5               ;MOV PUSHED INTO YELLOW ZONE
13009
13010  047224  010337  000336          8$: MOV     R3,@#336        ;RESTORE VECTOR 336
13011  047230  010506                      MOV     R5,SP           ;RESET SP
13012
13013                                  ;;*****************************************************************
13014                                  ;*TEST 663      YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
13015                                  ;;*****************************************************************
13016  047232                          TST663:
```

```
13017  047232  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13018  047234  012700  000663          MOV     #663,R0         ;;LOAD R0 WITH TEST NUMBER
13019  047240  013701  047272          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13020  047244  010605                  MOV     SP,R5           ;SAVE SP
13021  047246  012702  000376          MOV     #376,R2         ;R2 POINTS TO STACK
13022  047252  013704  000004          MOV     @#4,R4          ;SAVE T.O. VECTOR
13023  047256  012737  047310  000004  MOV     #4$,@#4 ;ON OVFLW - GO TO 4$
13024  047264  012706  000400          MOV     #400,SP         ;SET SP TO CAUSE OVFLW
13025  047270  000257                  CCC                     ;SCOPE SYNC
13026
13027  047272  010046          2$:     MOV     R0,-(SP)        ;FORCE STACK OVFLW - GO TO 4$
13028
13029  047274  010437  000004          MOV     R4,@#4          ;RESTORE T.O. VECTOR
13030  047300  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13031  047304  010506                  MOV     R5,SP           ;RESET SP FOR ERROR CALL
13032  047306  104005          3$:     ERROR   5               ;STACK OVFLW FAILED TO TRAP
13033
13034  047310  010437  000004  4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13035  047314  020012                  CMP     R0,(R2)         ;DID [R0] GET PUSHED?
13036  047316  001404                  BEQ     6$              ;BR IF YES
13037
13038  047320  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13039  047324  010506                  MOV     R5,SP           ;RESET SP FOR ERROR CALL
13040  047326  104005          5$:     ERROR   5               ;MOV FAILED TO PUSH IN YELLOW ZONE
13041
13042  047330  005706          6$:     TST     SP              ;[SP]=0?
13043  047332  001004                  BNE     8$              ;BR IF NOT
13044
13045  047334  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13046  047340  010506                  MOV     R5,SP           ;RESET SP FOR ERROR CALL
13047  047342  104005          7$:     ERROR   5               ;RED ZONE INSTEAD OF YELLOW ZONE
13048
13049  047344  010506          8$:     MOV     R5,SP           ;RESET SP
13050
13051                          ;;*****************************************************************
13052                          ;*TEST 664       YELLOW ZONE OVERFLOW TEST - (CMP  R0,-(SP))
13053                          ;;*****************************************************************
13054  047346                  TST664:
13055  047346  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13056  047350  012700  000664          MOV     #664,R0         ;;LOAD R0 WITH TEST NUMBER
13057  047354  013701  047402          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13058  047360  010605                  MOV     SP,R5           ;SAVE THE SP
13059  047362  013704  000004          MOV     @#4,R4          ;SAVE TRAP VECTOR
13060  047366  012737  047406  000004  MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13061  047374  012706  000400          MOV     #400,SP         ;SET SP TO PUSH INTO 'YELLOW ZONE'
13062  047400  000257                  CCC                     ;SCOPE SYNC
13063
13064  047402  020046          2$:     CMP     R0,-(SP)        ;TEST THE CMP - NO TRAP SHOULD OCCUR
13065
13066  047404  000406                  BR      6$              ;GO TO EXIT TEST
13067
13068  047406  010437  000004  4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13069  047412  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13070  047416  010506                  MOV     R5,SP           ;RESET THE SP
13071  047420  104005          3$:     ERROR   5               ;CMP CAUSED OVERFLOW TRAP
13072
```

```
13073  047422  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13074  047426  010506                          MOV     R5,SP           ;RESET THE SP
13075
13076                                  ;;**********************************************************************
13077                                  ;*TEST 665       YELLOW ZONE OVERFLOW TEST - (BIT  R0,-(SP))
13078                                  ;;**********************************************************************
13079  047430                          TST665:
13080  047430  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13081  047432  012700  000665                  MOV     #665,R0         ;;LOAD R0 WITH TEST NUMBER
13082  047436  013701  047464                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13083  047442  010605                          MOV     SP,R5           ;SAVE THE SP
13084  047444  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
13085  047450  012737  047470  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13086  047456  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO 'YELLOW ZONE'
13087  047462  000257                          CCC                     ;SCOPE SYNC
13088
13089  047464  030046          2$:     BIT     R0,-(SP)        ;TEST THE BIT - NO TRAP SHOULD OCCUR
13090
13091  047466  000406                  BR      6$              ;GO TO EXIT TEST
13092
13093  047470  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13094  047474  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13095  047500  010506                          MOV     R5,SP           ;RESET THE SP
13096  047502  104005          3$:     ERROR   5               ;BIT CAUSED OVERFLOW TRAP
13097
13098  047504  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13099  047510  010506                          MOV     R5,SP           ;RESET THE SP
13100
13101                                  ;;**********************************************************************
13102                                  ;*TEST 666       YELLOW ZONE OVERFLOW TEST - (TST  -(SP))
13103                                  ;;**********************************************************************
13104  047512                          TST666:
13105  047512  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13106  047514  012700  000666                  MOV     #666,R0         ;;LOAD R0 WITH TEST NUMBER
13107  047520  013701  047546                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13108  047524  010605                          MOV     SP,R5           ;SAVE THE SP
13109  047526  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
13110  047532  012737  047552  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13111  047540  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO 'YELLOW ZONE'
13112  047544  000257                          CCC                     ;SCOPE SYNC
13113
13114  047546  005746          2$:     TST     -(SP)           ;TEST THE TST - NO TRAP SHOULD OCCUR
13115
13116  047550  000406                  BR      6$              ;GO TO EXIT TEST
13117
13118  047552  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13119  047556  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13120  047562  010506                          MOV     R5,SP           ;RESET THE SP
13121  047564  104006          3$:     ERROR   6               ;TST CAUSED OVERFLOW TRAP
13122
13123  047566  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13124  047572  010506                          MOV     R5,SP           ;RESET THE SP
13125
13126                                  ;;**********************************************************************
13127                                  ;*TEST 667       ODD ADDRESS ERROR TEST - SUB  RA,(RB) - (RB) = ODD
13128                                  ;;**********************************************************************
```

```
13129  047574
13130  047574  000004                  TST667:
13131  047576  012700  000667                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13132  047602  013701  047636                  MOV     #667,R0         ;;LOAD R0 WITH TEST NUMBER
13133  047606  010605                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13134  047610  010737  001010                  MOV     SP,R5           ;SAVE SP
13135  047614  013704  000004          1$:     MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13136  047620  012737  047646  000004          MOV     @#4,R4          ;SAVE T.O. VECTOR
13137  047626  010506                          MOV     #4$,@#4 ;ON ODD ADDR ERROR - GO TO 4$
13138  047630  012702  000001                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13139  047634  000257                          MOV     #1,R2           ;R2 GETS ODD ADDRESS
13140                                          CCC                     ;SCOPE SYNC
13141  047636  160012                  2$:     SUB     R0,(R2)         ;FORCE ODD ADDR ERROR - GO TO 4$
13142
13143  047640  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13144  047644  104006                  3$:     ERROR   6               ;ODD ADDR FAILED TO TRAP
13145
13146  047646  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13147  047652  010506                          MOV     R5,SP           ;RESET SP
13148  047654  005037  000000                  CLR     @#0             ;CLR LOC. 0 JUST IN CASE
13149
13150                                  ;;*******************************************************************
13151                                  ;*TEST 670      TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
13152                                  ;;*******************************************************************
13153  047660                          TST670:
13154  047660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13155  047662  012700  000670                  MOV     #670,R0         ;;LOAD R0 WITH TEST NUMBER
13156  047666  013701  047710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13157  047672  012702  063317                  MOV     #MBUF1+1,R2     ;DEST ADDR=MBUF1+1 (ODD)
13158  047676  012737  047760  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13159
13160  047704  010205                          MOV     R2,R5           ;[R5] = DEST. ADDR
13161  047706  000257                          CCC                     ;SCOPE SYNC
13162
13163  047710  105435                  2$:     NEGB    @(R5)+          ;TEST DM=3 TRAP
13164
13165  047712  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13166
13167  047714  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = DEST. ADDR
13168  047720  013701  047726                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13169  047724  000257                          CCC                     ;SCOPE SYNC
13170
13171  047726  105455                  20$:    NEGB    @-(R5)          ;TEST DM=5 TRAP
13172
13173  047730  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13174
13175  047732  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13176  047734  013701  047742                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13177  047740  000257                          CCC                     ;SCOPE SYNC
13178
13179  047742  105475  000000          21$:    NEGB    @0(R5)          ;TEST DM=7 TRAP
13180
13181  047746  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13182
13183  047750  012737  061220  000004          MOV     #CERR,@#4       ;RESET T.O. VECTOR
13184  047756  000403                          BR      TST671          ;;GO TO SCOPE EXIT
```

```
13185
13186  047760  062716  000002          4$:     ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13187  047764  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13188
13189                                  ;;**********************************************************************
13190                                  ;*TEST 671       TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
13191                                  ;;**********************************************************************
13192  047766                          TST671:
13193  047766  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13194  047770  012700  000671                  MOV     #671,R0         ;;LOAD R0 WITH TEST NUMBER
13195  047774  013701  050016                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13196  050000  012702  063317                  MOV     #MBUF1+1,R2     ;[R2] = SOURCE ADDR. (ODD)
13197  050004  012737  050066  000004          MOV     #4$,@#4 ;GO TO 4$ ON TRAP
13198
13199  050012  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR.
13200  050014  000257                          CCC                     ;SCOPE SYNC
13201
13202  050016  113504                  2$:     MOVB    @(R5)+,R4       ;TEST SM=3
13203
13204  050020  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13205
13206  050022  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = SOURCE ADDR
13207  050026  013701  050034                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13208  050032  000257                          CCC                     ;SCOPE SYNC
13209
13210  050034  115504                  20$:    MOVB    @-(R5),R4       ;TEST SM=5
13211
13212  050036  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13213  050040  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR
13214  050042  013701  050050                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13215  050046  000257                          CCC                     ;SCOPE SYNC
13216
13217  050050  117504  000000          21$:    MOVB    @0(R5),R4       ;TEST SM=7
13218
13219  050054  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13220
13221  050056  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13222  050064  000403                          BR      TST672          ;;GO TO SCOPE EXIT
13223
13224  050066  062716  000002          4$:     ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
13225  050072  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13226
13227                                  ;;**********************************************************************
13228                                  ;*TEST 672       TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES
13229                                  ;;**********************************************************************
13230  050074                          TST672:
13231  050074  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13232  050076  012700  000672                  MOV     #672,R0         ;;LOAD R0 WITH TEST NUMBER
13233  050102  013701  050124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13234  050106  012702  050177                  MOV     #6$+3,R2        ;DEST ADDR = 6$+3 (ODD)
13235  050112  012737  050202  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13236
13237  050120  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13238  050122  000257                          CCC                     ;SCOPE SYNC
13239
13240  050124  000135                  2$:     JMP     @(R5)+          ;TEST JMP DM=3
```

```
13241
13242  050126  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 153
13243
13244  050130  012705  050177                  MOV     #6$+3,R5        ;[R5] = DEST ADDR
13245  050134  013701  050142                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13246  050140  000257                          CCC                     ;SCOPE SYNC
13247
13248  050142  000155                  20$:    JMP     @-(R5)          ;TEST JMP DM=5
13249
13250  050144  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 155
13251
13252  050146  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13253  050150  013701  050156                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13254  050154  000257                          CCC                     ;SCOPE SYNC
13255
13256  050156  000175  000000          21$:    JMP     @0(R5)          ;TEST JMP DM=7
13257
13258  050162  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13259
13260  050164  012737  061220  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13261  050172  000420                          BR      TST673          ;;GO TO SCOPE EXIT
13262
13263  050174  000000                  6$:     HALT                    ;CATASTOPHIC ERROR - [PC] QUESTIONABLE.
13264  050176  000000                          HALT                    ;RESTART PROGRAM - DO NOT CONTINUE.
13265  050200  000000                          HALT
13266
13267  050202  032716  000001          4$:     BIT     #1,(SP)         ;TRAP DUE TO ODD PC?
13268  050206  001003                          BNE     8$              ;BR IF YES
13269  050210  062716  000002                  ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13270  050214  000002                          RTI                     ;RETURN TO NEXT SUB TEST
13271
13272  050216  011603                  8$:     MOV     (SP),R3         ;GET ODD PC OFF STACK INTO R3
13273  050220  062706  000004                  ADD     #4,SP           ;FIX SP
13274
13275  050224  104007                  9$:     ERROR   7               ;PC TRAPPED WITH ODD ADDRESS
13276
13277  050226  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13278
13279                                  ;;**********************************************************************
13280                                  ;*TEST 673       TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
13281                                  ;;**********************************************************************
13282  050234                          TST673:
13283  050234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13284  050236  012700  000673                  MOV     #673,R0         ;;LOAD R0 WITH TEST NUMBER
13285  050242  013701  050266                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13286  050246  012737  050400  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13287  050254  010605                          MOV     SP,R5           ;SAVE SP
13288  050256  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13289
13290  050262  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13291  050264  000257                          CCC                     ;SCOPE SYNC
13292
13293  050266  005016                  2$:     CLR     (SP)            ;TEST DM1 - SHOULD SPRING TRAP
13294
13295  050270  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13296  050274  010506                          MOV     R5,SP           ;RESET SP
```

```
13297  050276  104006                  3$:     ERROR   6               ;DM1 FAILED TO CAUSE OVERFLOW TRAP
13298
13299  050300  013701  050310                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
13300  050304  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13301  050306  000257                          CCC                     ;SCOPE SYNC
13302
13303  050310  005026                  20$:    CLR     (SP)+           ;TEST DM2 - SHOULD SPRING TRAP
13304
13305  050312  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13306  050316  010506                          MOV     R5,SP           ;RESET SP
13307  050320  104006                  5$:     ERROR   6               ;DM2 FAILED TO CAUSE OVERFLOW TRAP
13308
13309  050322  013701  050332                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
13310  050326  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13311  050330  000257                          CCC                     ;SCOPE SYNC
13312
13313  050332  005046                  21$:    CLR     -(SP)           ;TEST DM4 - SHOULD SPRING TRAP
13314
13315  050334  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13316  050340  010506                          MOV     R5,SP           ;RESET SP
13317  050342  104006                  7$:     ERROR   6               ;DM4 FAILED TO CAUSE OVERFLOW TRAP
13318
13319  050344  013701  050354                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
13320  050350  010206                          MOV     R2,SP           ;SET SP TO CAUSE ERROR
13321  050352  000257                          CCC                     ;SCOPE SYNC
13322
13323  050354  005066  000000          22$:    CLR     0(SP)           ;TEST DM6 - SHOULD SPRING TRAP
13324
13325  050360  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13326  050364  010506                          MOV     R5,SP           ;RESET SP
13327  050366  104006                  9$:     ERROR   6               ;DM6 FAILED TO CAUSE OVERFLOW TRAP
13328
13329  050370  012737  061220  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13330  050376  000407                          BR      TST674          ;;GO TO SCOPE EXIT
13331
13332  050400  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC OFF STACK
13333  050402  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13334  050406  010506                          MOV     R5,SP           ;RESET SP
13335  050410  005046                          CLR     -(SP)           ;PUSH NEW PS ON STACK
13336  050412  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC ON STACK
13337  050414  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13338
13339                                  ;;*********************************************************************
13340                                  ;*TEST 674       TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
13341                                  ;;*********************************************************************
13342  050416                          TST674:
13343  050416  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13344  050420  012700  000674                  MOV     #674,R0         ;;LOAD R0 WITH TEST NUMBER
13345  050424  013701  050450                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13346  050430  012737  050562  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLW TRAP
13347  050436  010605                          MOV     SP,R5           ;SAVE SP
13348  050440  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13349
13350  050444  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13351  050446  000257                          CCC                     ;SCOPE SYNC
13352
```

```
13353  050450  010016                2$:     MOV     R0,(SP)         ;TEST MOV DM1 - SHOULD SPRING TRAP
13354
13355  050452  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13356  050456  010506                        MOV     R5,SP           ;RESET SP
13357  050460  104006                3$:     ERROR   6               ;MOV DM1 FAILED TO SPRING TRAP
13358
13359  050462  013701  050472                MOV     @#20$,R1        ;[R1] = TEST INSTR.
13360  050466  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13361  050470  000257                        CCC                     ;SCOPE SYNC
13362
13363  050472  010026                20$:    MOV     R0,(SP)+        ;TEST MOV DM2 - SHOULD SPRING TRAP
13364
13365  050474  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13366  050500  010506                        MOV     R5,SP           ;RESET SP
13367  050502  104006                5$:     ERROR   6               ;MOV DM2 FAILED TO SPRING TRAP
13368
13369  050504  013701  050514                MOV     @#21$,R1        ;[R1] = TEST INSTR.
13370  050510  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13371  050512  000257                        CCC                     ;SCOPE SYNC
13372
13373  050514  010046                21$:    MOV     R0,-(SP)        ;TEST MOV DM4 - SHOULD SPRING TRAP
13374
13375  050516  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13376  050522  010506                        MOV     R5,SP           ;RESET SP
13377  050524  104006                7$:     ERROR   6               ;MOV DM4 FAILED TO SPRING TRAP
13378
13379  050526  013701  050536                MOV     @#22$,R1        ;[R1] = TEST INSTR.
13380  050532  010206                        MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13381  050534  000257                        CCC                     ;SCOPE SYNC
13382
13383  050536  010066  000000        22$:    MOV     R0,0(SP)        ;TEST MOV DM6 - SHOULD SPRING TRAP
13384
13385  050542  010637  001074                MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13386  050546  010506                        MOV     R5,SP           ;RESET SP
13387  050550  104006                9$:     ERROR   6               ;MOV DM6 FAILED TO CAUSE OVFLW TRAP
13388
13389  050552  012737  061220  000004        MOV     #BERR,@#4       ;RESET T.O. VECTOR
13390  050560  000407                        BR      TST675          ;;GO TO SCOPE EXIT
13391
13392  050562  011604                4$:     MOV     (SP),R4         ;GET RETURN PC
13393  050564  062704  000010                ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13394  050570  010506                        MOV     R5,SP           ;RESET SP
13395  050572  005046                        CLR     -(SP)           ;PUSH NEW PSW
13396  050574  010446                        MOV     R4,-(SP)        ;PUSH RETURN PC
13397  050576  000002                        RTI                     ;RETURN TO NEXT SUB-TEST
13398
13399                                ;;*********************************************************************
13400                                ;*TEST 675       TEST THAT JSR CAN CAUSE OVERFLOW TRAP
13401                                ;;*********************************************************************
13402  050600                        TST675:
13403  050600  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13404  050602  012700  000675                MOV     #675,R0         ;;LOAD R0 WITH TEST NUMBER
13405  050606  013701  050630                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13406  050612  012737  050652  000004        MOV     #4$,@#4 ;GO TO 4$ ON OVERFLOW ERROR
13407  050620  010605                        MOV     SP,R5           ;SAVE SP
13408  050622  012706  000400                MOV     #400,SP         ;SET THE SP TO CAUSE TRAP
```

```
13409  050626  000257                          CCC                     ;SCOPE SYNC
13410
13411  050630  004737  050656          2$:     JSR     PC,@#6$ ;TEST JSR - SHOULD SPRING TRAP
13412
13413  050634  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13414  050640  010506                          MOV     R5,SP           ;RESET SP
13415  050642  104005                  3$:     ERROR   5               ;JSR PUSH DID NOT SPRING OVFL TRAP
13416
13417  050644  000410                          BR      8$              ;GO TO SCOPE EXIT
13418
13419  050646  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13420  050652  010506                  4$:     MOV     R5,SP           ;RESET SP
13421  050654  000404                          BR      8$              ;GO EXIT TEST - ALL OK
13422
13423  050656  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13424  050662  010506                          MOV     R5,SP           ;RESET SP
13425  050664  104005                  5$:     ERROR   5               ;JSR PUSH FAILED TO SPRING OVFLW TRAP
13426
13427  050666  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13428
13429                                  ;;**********************************************************************
13430                                  ;*TEST 676       TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
13431                                  ;;**********************************************************************
13432  050674                          TST676:
13433  050674  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13434  050676  012700  000676                  MOV     #676,R0         ;;LOAD R0 WITH TEST NUMBER
13435  050702  013701  050736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13436  050706  013704  000014                  MOV     @#14,R4         ;SAVE BREAK POINT TRAP VECTOR
13437  050712  010605                          MOV     SP,R5           ;SAVE SP
13438  050714  012737  050752  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13439  050722  012737  050756  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13440  050730  012706  000400                  MOV     #400,SP         ;SET UP SP TO CAUSE OVFLW ON 1ST PUSH
13441  050734  000257                          CCC                     ;SCOPE SYNC
13442
13443  050736  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13444
13445  050740  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13446  050744  010506                          MOV     R5,SP           ;RESET SP
13447  050746  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
13448
13449  050750  000406                          BR      8$              ;GO TO SCOPE EXIT
13450
13451  050752  010506                  4$:     MOV     R5,SP           ;RESET SP
13452  050754  000404                          BR      8$              ;GO EXIT - ALL OK
13453
13454  050756  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13455  050762  010506                          MOV     R5,SP           ;RESET SP
13456  050764  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13457
13458  050766  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13459  050774  010437  000014                  MOV     R4,@#14
13460
13461                                  ;;**********************************************************************
13462                                  ;*TEST 677       TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
13463                                  ;;**********************************************************************
13464  051000                          TST677:
```

```
13465  051000  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13466  051002  012700  000677            MOV     #677,R0         ;;LOAD R0 WITH TEST NUMBER
13467  051006  013701  051042            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13468  051012  013704  000014            MOV     @#14,R4         ;SAVE BPT VECTOR
13469  051016  010605                    MOV     SP,R5           ;SAVE SP
13470  051020  012737  051056  000004    MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLOW
13471  051026  012737  051062  000014    MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13472  051034  012706  000402            MOV     #402,SP         ;SET SP TO CAUSE TRAP ON 2ND PUSH
13473  051040  000257                    CCC                     ;SCOPE SYNC
13474
13475  051042  000003            2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13476
13477  051044  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13478  051050  010506                    MOV     R5,SP           ;RESET SP
13479  051052  104005            3$:     ERROR   5               ;BPT FAILED TO TRAP
13480
13481  051054  000406                    BR      8$              ;GO TO SCOPE EXIT
13482
13483  051056  010506            4$:     MOV     R5,SP           ;RESET SP
13484  051060  000404                    BR      8$              ;GO EXIT - ALL OK
13485
13486  051062  010637  001074    6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13487  051066  010506                    MOV     R5,SP           ;RESET SP
13488  051070  104005            5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13489
13490  051072  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13491  051100  010437  000014                  MOV     R4,@#14
13492
13493                                    ;;******************************************************************
13494                                    ;*TEST 700      ILLEGAL INSTRUCTION TEST - JSR RN,%R
13495                                    ;;******************************************************************
13496  051104                            TST700:
13497  051104  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13498  051106  012700  000700                    MOV     #700,R0         ;;LOAD R0 WITH TEST NUMBER
13499  051112  013701  051146                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13500  051116  010605                            MOV     SP,R5           ;SAVE SP
13501  051120  010737  001010                    MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13502  051124  013704  000004            1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13503  051130  012737  051156  000004            MOV     #4$,@#4 ;ILLEGAL INSTR. TRAP GOES TO 4$
13504  051136  010506                            MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13505  051140  012702  051154                    MOV     #3$,R2          ;IN CASE JSR JUMPS TO [R2]
13506  051144  000257                            CCC                     ;SCOPE SYNC
13507
13508  051146  004302                    2$:     JSR     R3,R2           ;JSR MODE 0 FORCES TRAP - GO TO 4$
13509
13510  051150  010437  000004                    MOV     R4,@#4          ;RESTORE T.O. VECTOR
13511  051154  104005                    3$:     ERROR   5               ;JSR FAILED TO SPRING TRAP
13512
13513  051156  010437  000004            4$:     MOV     R4,@#4          ;RESTORE VECTOR
13514  051162  010506                            MOV     R5,SP           ;RESET SP
13515
13516                                    ;;******************************************************************
13517                                    ;*TEST 701      ILLEGAL INSTRUCTION TEST - JMP %R
13518                                    ;;******************************************************************
13519  051164                            TST701:
13520  051164  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
13521  051166  012700  000701              MOV     #701,R0          ;;LOAD R0 WITH TEST NUMBER
13522  051172  013701  051226              MOV     @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
13523  051176  010605                      MOV     SP,R5            ;SAVE SP
13524  051200  010737  001010              MOV     PC,@#$LPERR      ;SET ERROR LOOP ADDRESS
13525  051204  013704  000004      1$:     MOV     @#4,R4           ;SAVE VECTOR POINTER AT LOC. 4
13526  051210  012737  051236  000004      MOV     #4$,@#4 ;ON TRAP - GO TO 4$
13527  051216  010506                      MOV     R5,SP            ;RESET SP FOR ERROR LOOP
13528  051220  012702  051234              MOV     #3$,R2           ;IN CASE IT JUMPS TO ADDR IN RN
13529  051224  000257                      CCC                      ;SCOPE SYNC
13530
13531  051226  000102              2$:     JMP     R2               ;JMP MODE 0 FORCES TRAP - GO TO 4$
13532
13533  051230  010437  000004              MOV     R4,@#4           ;RESTORE VECTOR POINTER AT LOC. 4
13534  051234  104005              3$:     ERROR   5                ;ILLEGAL INSTR TRAP FAILED
13535
13536  051236  010437  000004      4$:     MOV     R4,@#4           ;RESTORE VECTOR POINTER AT LOC. 4
13537  051242  010506                      MOV     R5,SP            ;RESET SP
13538
13539                                       ;;*********************************************************************
13540                                       ;*TEST 702       BUS TIMEOUT TRAP TEST - TST (R)
13541                                       ;;*********************************************************************
13542  051244                              TST702:
13543  051244  000004                      SCOPE                    ;CALL THE SCOPE LOOP UTILITY
13544  051246  012700  000702              MOV     #702,R0          ;;LOAD R0 WITH TEST NUMBER
13545  051252  013701  051306              MOV     @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
13546  051256  010605                      MOV     SP,R5            ;SAVE SP
13547  051260  010737  001010              MOV     PC,@#$LPERR      ;SET ERROR LOOP ADDRESS
13548  051264  013704  000004      1$:     MOV     @#4,R4           ;SAVE ORIGINAL T.O. VECTOR POINTER
13549  051270  012737  051316  000004      MOV     #4$,@#4 ;ON T.O. TRAP - GO TO 4$
13550  051276  012702  160000              MOV     #160000,R2       ;ADDRESS CAUSES T.O.
13551  051302  010506                      MOV     R5,SP            ;RESET SP FOR ERROR LOOP
13552  051304  000257                      CCC                      ;SCOPE SYNC
13553
13554  051306  005712              2$:     TST     (R2)             ;FORCE T.O. TRAP - GO TO 4$
13555
13556  051310  010437  000004              MOV     R4,@#4           ;RESTORE T.O. VECTOR
13557  051314  104005              3$:     ERROR   5                ;TIMEOUT TRAP FAILED
13558  051316  010437  000004      4$:     MOV     R4,@#4           ;RESTORE T.O. VECTOR
13559  051322  010506                      MOV     R5,SP            ;RESET SP
13560
13561                                       ;;*********************************************************************
13562                                       ;*TEST 703       'T' BIT TRAP TEST
13563                                       ;;*********************************************************************
13564  051324                              TST703:
13565  051324  000004                      SCOPE                    ;CALL THE SCOPE LOOP UTILITY
13566  051326  012700  000703              MOV     #703,R0          ;;LOAD R0 WITH TEST NUMBER
13567  051332  013701  051370              MOV     @#2$,R1          ;LOAD R1 WITH TEST INSTRUCTION WORD
13568  051336  010605                      MOV     SP,R5            ;SAVE SP
13569  051340  010737  001010              MOV     PC,@#$LPERR      ;SET ERROR LOOP ADDRESS
13570  051344  010506              1$:     MOV     R5,SP            ;RESET SP FOR ERROR LOOP
13571  051346  012737  051376  000014      MOV     #4$,@#14         ;GO TO 4$ WHEN 'T' TRAP SPRUNG
13572  051354  012746  000020              MOV     #20,-(SP)        ;SET 'T' BIT ON STACK
13573  051360  012746  051370              MOV     #2$,-(SP)        ;SET UP NEW PC ON STACK
13574  051364  000257                      CCC                      ;SCOPE SYNC
13575  051366  000006                      RTT                      ;TURN ON 'T' BIT - GO TO 2$
13576
```

```
13577  051370  005700                  2$:     TST     R0              ;SPRING 'T'' BIT TRAP - GO TO 4$
13578
13579  051372  104005                  3$:     ERROR   5               ;NO 'T'' BIT TRAP OCCURRED
13580
13581  051374  000405                          BR      6$              ;GO EXIT
13582
13583  051376  032766  000020  000002  4$:     BIT     #20,2(SP)       ;'T'' BIT SET IN OLD PSW?
13584  051404  001001                          BNE     6$              ;BR IF YES
13585
13586  051406  104001                  5$:     ERROR   1               ;#T# BIT NOT SAVED ON STACK
13587
13588  051410  012737  000016  000014  6$:     MOV     #16,@#14        ;RESTORE 'T'' BIT TRAP CATCHER
13589  051416  005037  000016                  CLR     @#16
13590  051422  010506                          MOV     R5,SP           ;RESET SP
13591
13592                                          ;;**********************************************************************
13593                                          ;*TEST 704      TEST PUSH INTO PSW WITH [SP] = 000000
13594                                          ;THESE NEXT TWO TESTS VERIFY THAT A 'RED ZONE'' TRAP OCCURS IF A
13595                                          ;PUSH IS ATTEMPTED WITH THE [SP] INITIALLY EQUAL TO 000000,177572,
13596                                          ;;**********************************************************************
13597  051424                          TST704:
13598  051424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13599  051426  012700  000704                  MOV     #704,R0         ;;LOAD R0 WITH TEST NUMBER
13600  051432  013701  051456                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13601  051436  010605                          MOV     SP,R5           ;SAVE THE SP
13602  051440  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13603  051444  012737  051500  000004          MOV     #4$,@#4 ;'RED ZONE'' TRAP GOES TO 4$
13604  051452  005006                          CLR     SP              ;MAKE SP = 000000
13605  051454  000257                          CCC                     ;SCOPE SYNC
13606
13607  051456  012746  007777          2$:     MOV     #7777,-(SP)     ;ATTEMPT PUSH INTO PSW - SHOULD CAUSE
13608                                                                  ;'RED ZONE'' TRAP TO BE SPRUNG
13609
13610  051462  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13611  051466  005004                          CLR     R4              ;[R4] = S / B SP
13612  051470  010603                          MOV     SP,R3           ;[R3] = WAS SP
13613  051472  010506                          MOV     R5,SP           ;RESET THE SP
13614  051474  104003                  3$:     ERROR   3               ;TRAP NOT SPRUNG
13615  051476  000414                          BR      TST705          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13616
13617  051500  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13618  051504  001406                          BEQ     6$              ;BR IF YES
13619
13620  051506  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13621  051512  005004                          CLR     R4              ;[R4]= S / B SP
13622  051514  010603                          MOV     SP,R3           ;[R3] = WAS SP
13623  051516  010506                          MOV     R5,SP           ;RESET THE SP
13624  051520  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13625
13626  051522  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
13627  051524  010437  000004                  MOV     R4,@#4          ;RESTORE BERR VECTOR
13628
13629                                          ;;**********************************************************************
13630                                          ;*TEST 705      TEST PUSH INTO SR WITH [SP] = 177572
13631                                          ;;**********************************************************************
13632  051530                          TST705:
```

```
13633  051530  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13634  051532  012700  000705                MOV     #705,R0         ;;LOAD R0 WITH TEST NUMBER
13635  051536  013701  051564                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13636  051542  010605                        MOV     SP,R5           ;SAVE THE SP
13637  051544  013704  000004                MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13638  051550  012737  051606  000004        MOV     #4$,@#4 ;''RED ZONE'' TRAP GOES TO 4$
13639  051556  012706  177572                MOV     #177572,SP      ;MAKE SP=177572
13640  051562  000257                        CCC                     ;SCOPE SYNC
13641
13642  051564  012746  177777        2$:     MOV     #-1,-(SP)       ;ATTEMPT PUSH INTO SR - SHOULD CAUSE
13643                                                                 ;''RED ZONE'' TRAP TO BE SPRUNG
13644
13645  051570  010437  000004                MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13646  051574  005004                        CLR     R4              ;[R4] = S / B SP
13647  051576  010603                        MOV     SP,R3           ;[R3] = WAS SP
13648  051600  010506                        MOV     R5,SP           ;RESET THE SP
13649  051602  104003        3$:     ERROR   3                       ;TRAP NOT SPRUNG
13650  051604  000414                        BR      TST706          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13651
13652  051606  022706  000000        4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13653  051612  001406                        BEQ     6$              ;BR IF YES
13654
13655  051614  010437  000004                MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13656  051620  005004                        CLR     R4              ;[R4]= S / B SP
13657  051622  010603                        MOV     SP,R3           ;[R3] = WAS SP
13658  051624  010506                        MOV     R5,SP           ;RESET THE SP
13659  051626  104003        5$:     ERROR   3                       ;TRAP SPRUNG BUT NOT RED ZONE
13660
13661  051630  010506        6$:     MOV     R5,SP           ;FIX UP THE SP
13662  051632  010437  000004                MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13663
13664                        ;;*******************************************************************
13665                        ;*TEST 706       TEST PUSH INTO SLR WITH [SP] = 177776
13666                        ;;*******************************************************************
13667  051636                TST706:
13668  051636  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13669  051640  012700  000706                MOV     #706,R0         ;;LOAD R0 WITH TEST NUMBER
13670  051644  013701  051672                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13671  051650  010605                        MOV     SP,R5           ;SAVE THE SP
13672  051652  013704  000004                MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13673  051656  012737  051714  000004        MOV     #4$,@#4 ;''RED ZONE'' TRAP GOES TO 4$
13674  051664  012706  177776                MOV     #177776,SP      ;MAKE SP=177776
13675  051670  000257                        CCC                     ;SCOPE SYNC
13676
13677  051672  012746  000200        2$:     MOV     #200,-(SP)      ;ATTEMPT PUSH INTO SLR - SHOULD CAUSE
13678                                                                 ;''RED ZONE'' TRAP TO BE SPRUNG
13679
13680  051676  010437  000004                MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13681  051702  005004                        CLR     R4              ;[R4] = S / B SP
13682  051704  010603                        MOV     SP,R3           ;[R3] = WAS SP
13683  051706  010506                        MOV     R5,SP           ;RESET THE SP
13684  051710  104003        3$:     ERROR   3                       ;TRAP NOT SPRUNG
13685  051712  000414                        BR      TST707          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13686
13687  051714  022706  000000        4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13688  051720  001406                        BEQ     6$              ;BR IF YES
```

```
13689
13690  051722  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13691  051726  005004                  CLR     R4              ;[R4]= S / B SP
13692  051730  010603                  MOV     SP,R3           ;[R3] = WAS SP
13693  051732  010506                  MOV     R5,SP           ;RESET THE SP
13694  051734  104003          5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13695
13696  051736  010506          6$:     MOV     R5,SP           ;FIX UP THE SP
13697  051740  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13698
13699                          ;;*******************************************************************
13700                          ;*TEST 707      RSVD INSTRUCTION TEST - 000007 THRU 000077
13701                          ;;*******************************************************************
13702  051744                  TST707:
13703  051744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13704  051746  012700  000707          MOV     #707,R0         ;;LOAD R0 WITH TEST NUMBER
13705  051752  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13706  051754  012737  052012  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13707  051762  005037  000012          CLR     @#12
13708  051766  012701  000007          MOV     #7,R1           ;SET UP FIRST ONE IN GROUP
13709  051772  010737  001010          MOV     PC,@#$LPERR     ;ONLY LOOP ON BAD OP CODE
13710  051776  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13711  052000  010137  052006          MOV     R1,@#2$ ;LOAD NEW INSTR
13712  052004  000257                  CCC                     ;SCOPE SYNC
13713
13714  052006  000007          2$:     000007                  ;TEST THE RSVD INSTR - THIS LOCATION
13715                                                          ;GETS CHANGED EACH PASS THROUGH
13716
13717  052010  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13718
13719  052012  005201          4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13720  052014  022701  000100          CMP     #100,R1         ;AT END OF THIS GROUP ??
13721  052020  001366                  BNE     1$              ;BR IF NOT
13722
13723  052022  010506                  MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13724  052024  012737  051752  001010  MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13725                          ;;*******************************************************************
13726                          ;*TEST 710      RSVD INSTRUCTION TEST - 000210 THRU 000237
13727                          ;;*******************************************************************
13728  052032                  TST710:
13729  052032  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13730  052034  012700  000710          MOV     #710,R0         ;;LOAD R0 WITH TEST NUMBER
13731  052040  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13732  052042  012737  052100  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13733  052050  005037  000012          CLR     @#12
13734  052054  012701  000210          MOV     #210,R1         ;SET UP FIRST ONE IN GROUP
13735  052060  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13736  052064  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13737  052066  010137  052074          MOV     R1,@#2$ ;LOAD NEW INSTR
13738  052072  000257                  CCC                     ;SCOPE SYNC
13739
13740  052074  000210          2$:     000210                  ;TEST THE RSVD INSTR - THIS LOCATION
13741                                                          ;GETS CHANGED EACH PASS THROUGH
13742
13743  052076  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13744
```

```
13745  052100  005201                    4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13746  052102  022701  000240                    CMP     #240,R1         ;AT END OF THIS GROUP ??
13747  052106  001366                            BNE     1$              ;BR IF NOT
13748
13749  052110  010506                            MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13750  052112  012737  052040  001010            MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13751
13752                                    ;;*******************************************************************
13753                                    ;*TEST 711       RSVD INSTRUCTION TEST - 007000 THRU 007777
13754                                    ;;*******************************************************************
13755  052120                            TST711:
13756  052120  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13757  052122  012700  000711                    MOV     #711,R0         ;;LOAD R0 WITH TEST NUMBER
13758  052126  010605                    5$:     MOV     SP,R5           ;SAVE THE SP
13759  052130  012737  052166  000010            MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13760  052136  005037  000012                    CLR     @#12
13761  052142  012701  007000                    MOV     #7000,R1        ;SET UP FIRST ONE IN GROUP
13762  052146  010737  001010                    MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13763  052152  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13764  052154  010137  052162                    MOV     R1,@#2$ ;LOAD NEW INSTR
13765  052160  000257                            CCC                     ;SCOPE SYNC
13766
13767  052162  007000                    2$:     007000                  ;TEST THE RSVD INSTR - THIS LOCATION
13768                                            ;GETS CHANGED EACH PASS THROUGH
13769
13770  052164  104005                    3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13771
13772  052166  005201                    4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13773  052170  022701  010000                    CMP     #10000,R1       ;AT END OF THIS GROUP ??
13774  052174  001366                            BNE     1$              ;BR IF NOT
13775
13776  052176  010506                            MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13777  052200  012737  052126  001010            MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13778
13779                                    ;;*******************************************************************
13780                                    ;*TEST 712       RSVD INSTRUCTION TEST - 075000 THRU 076777
13781                                    ;;*******************************************************************
13782  052206                            TST712:
13783  052206  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13784  052210  012700  000712                    MOV     #712,R0         ;;LOAD R0 WITH TEST NUMBER
13785  052214  010605                    5$:     MOV     SP,R5           ;SAVE THE SP
13786  052216  012737  052256  000010            MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13787  052224  005037  000012                    CLR     @#12
13788  052230  012701  075000                    MOV     #75000,R1       ;SET UP FIRST ONE IN GROUP
13789  052234  010737  001010                    MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13790  052240  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13791  052242  010137  052250                    MOV     R1,@#2$ ;LOAD NEW INSTR
13792  052246  000257                            CCC                     ;SCOPE SYNC
13793
13794  052250  075000                    2$:     75000                   ;TEST THE RSVD INSTR - THIS LOCATION
13795                                            ;GETS CHANGED EACH PASS THROUGH
13796
13797  052252  000240                            NOP                     ;IN CASE NON TRAPPING INSTR IS TWO WORDS
13798  052254  104005                    3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13799
13800  052256  005201                    4$:     INC     R1              ;GENERATE NEW RSVD INSTR
```

```
13801  052260  022701  076600            MED:    CMP     #MED,R1         ;MED INSTRUCTION?
13802  052264  001774                            BEQ     4$              ;BR IF YES--SKIP IT.
13803  052266  022701  077000                    CMP     #077000,R1      ;AT END OF THIS GROUP ??
13804  052272  001362                            BNE     1$              ;BR IF NOT
13805
13806  052274  010506                            MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13807  052276  012737  052214  001010            MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13808
13809                                    ;;**********************************************************************
13810                                    ;*TEST 713       RSVD INSTRUCTION TEST - 106400 THRU 107777
13811                                    ;;**********************************************************************
13812  052304                            TST713:
13813  052304  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13814  052306  012700  000713                    MOV     #713,R0         ;;LOAD R0 WITH TEST NUMBER
13815  052312  010605                    5$:     MOV     SP,R5           ;SAVE THE SP
13816  052314  012737  052352  000010            MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13817  052322  005037  000012                    CLR     @#12
13818  052326  012701  106400                    MOV     #106400,R1      ;SET UP FIRST ONE IN GROUP
13819  052332  010737  001010                    MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13820  052336  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13821  052340  010137  052346                    MOV     R1,@#2$ ;LOAD NEW INSTR
13822  052344  000257                            CCC                     ;SCOPE SYNC
```

```
13823
13824  052346  106400                  2$:     106400                  ;TEST THE RSVD INSTR - THIS LOCATION
13825                                                                   ;GETS CHANGED EACH PASS THROUGH
13826
13827  052350  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13828
13829  052352  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13830  052354  022701  106500                  CMP     #106500,R1      ;MFPD INSTRUCTION ??
13831  052360  001002                          BNE     10$             ;BR IF NOT
13832  052362  012701  106700                  MOV     #106700,R1      ;SKIP MFPD AND MTPD INSTRUCTIONS
13833  052366  022701  110000          10$:    CMP     #110000,R1      ;AT END OF THIS GROUP ??
13834  052372  001361                          BNE     1$              ;BR IF NOT
13835
13836  052374  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13837  052376  012737  052312  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13838  052404  012737  061122  000010          MOV     #RSERR,@#10     ;RESTORE RSVD INSTR VECTOR
13839  052412  012737  000340  000012          MOV     #340,@#12
13840  052420  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13841
13842                                  ;THIS NEXT GROUP OF SEQUENTIAL TESTS VERIFIES THAT A 'T' BIT
13843                                  ;TRAP CAN BE SERVICED IN EACH MICROWORD THAT DOES A 'BUT SERVICE'
13844                                  ;EACH ROUTINE ENTERS THE TRAP MICROUTINE WHEN THE TRAP IS SPRUNG
13845
13846  052422  012737  061070  000014  TSET:   MOV     #TBSER,@#14     ;SET UP THE 'T' BIT TRAP VECTOR
13847  052430  012737  000340  000016          MOV     #340,@#16       ;PRIORITY 7
13848
13849                                  ;;*********************************************************************
13850                                  ;*TEST 714      BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
13851                                  ;'INSTAB' (INSTRUCTION TABLE) CONTAINS ALL ONE WORD INSTRUCTIONS
13852                                  ;THAT TEST A 'BUT SERVICE' IN A UNIQUE ROM LOCATION.  THE TABLE MUST
13853                                  ;BE TERMINATED WITH A 0 ENTRY.
13854                                  ;;*********************************************************************
13855  052436                          TST714:
13856  052436  012700  000714                  MOV     #714,R0         ;;LOAD R0 WITH TEST NUMBER
13857  052442  010605                  6$:     MOV     SP,R5           ;SAVE THE SP
13858  052444  012704  063636                  MOV     #INSTAB,R4      ;PUT POINTER TO TABLE IN R4
13859  052450  012401                  4$:     MOV     (R4)+,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13860  052452  001422                          BEQ     5$              ;EXIT TEST IF END OF TABLE
13861  052454  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ON FAILING INSTRUCTION ONLY
13862  052460  010137  052512          1$:     MOV     R1,@#2$         ;STORE TEST INSTRUCTION TO BE EXECUTED
13863  052464  012702  063312                  MOV     #MBUF0,R2       ;IN CASE DM1 DEST--(R2)
13864  052470  012703  063316                  MOV     #MBUF1,R3       ;IN CASE SM1--(R3)
13865  052474  010506                          MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13866  052476  012746  000020                  MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13867  052502  012746  052512                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13868  052506  000257                          CCC                     ;SCOPE SYNC
13869  052510  000006                          RTT                     ;SET 'T' BIT - GO TO 2$
13870
13871  052512  000240          2$:     NOP                             ;INSTRUCTION FROM TABLE IS STORED HERE AND
13872                                                                   ;SHOULD SPRING TRAP
13873
13874  052514  104005                  3$:     ERROR   5               ;BUT SERVICE FAILED
13875
13876  052516  000754                          BR      4$              ;GET NEXT INSTRUCTION FOR BUT SERVICE TEST
13877  052520  012737  052442  001010  5$:     MOV     #6$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13878
```

```
13879                                ;;*****************************************************************
13880                                ;*TEST 715      BUT SERVICE TEST - (RTI)
13881                                ;;*****************************************************************
13882  052526                        TST715:
13883  052526  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13884  052530  012700  000715               MOV     #715,R0         ;;LOAD R0 WITH TEST NUMBER
13885  052534  013701  052552               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13886  052540  012746  000020               MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13887  052544  012746  052554               MOV     #3$,-(SP)       ;MAKE NEW PC = 3$
13888  052550  000257                       CCC                     ;SCOPE SYNC
13889
13890  052552  000002               2$:     RTI                     ;INSTRUCTION SHOULD SPRING TRAP
13891
13892  052554  104005               3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13893
13894                                ;;*****************************************************************
13895                                ;*TEST 716      BUT SERVICE TEST - (JSR  %R,@A)
13896                                ;;*****************************************************************
13897  052556                        TST716:
13898  052556  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13899  052560  012700  000716               MOV     #716,R0         ;;LOAD R0 WITH TEST NUMBER
13900  052564  013701  052634               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13901                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT14
13902  052570  032737  040000  063234       BIT     #BIT14,@#BPTLOC ;BREAKPOINT HALT SET ??
13903  052576  001401                       BEQ     .+4             ;BR IF NOT
13904  052600  000000                       HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
13905  052602  010605                       MOV     SP,R5           ;SAVE THE SP
13906  052604  010737  001010               MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13907  052610  010506               1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13908  052612  012737  052640  063316       MOV     #3$,@#MBUF1     ;SET UP POINTER--DEST ADDR = 3$ FOR JSR
13909  052620  012746  000020               MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13910  052624  012746  052634               MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13911  052630  000257                       CCC                     ;SCOPE SYNC
13912  052632  000006                       RTT                     ;SET 'T' BIT - GO TO 2$
13913
13914  052634  004777  010456       2$:     JSR     PC,@MBUF1       ;INSTRUCTION SHOULD SPRING TRAP
13915
13916  052640  104005               3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13917
13918  052642  010506                       MOV     R5,SP           ;RESTORE SP IF ALL OK OR NOT LOOPING
13919                                ;;*****************************************************************
13920                                ;*TEST 717      BUT SERVICE TEST - (JMP A)
13921                                ;;*****************************************************************
13922  052644                        TST717:
13923  052644  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13924  052646  012700  000717               MOV     #717,R0         ;;LOAD R0 WITH TEST NUMBER
13925  052652  013701  052672               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13926  052656  012746  000020               MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13927  052662  012746  052672               MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13928  052666  000257                       CCC                     ;SCOPE SYNC
13929  052670  000006                       RTT                     ;SET 'T' BIT - GO TO 2$
13930
13931  052672  000167  000000       2$:     JMP     3$              ;JMP INSTRUCTION SHOULD SPRING TRAP
13932
13933  052676  104005               3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13934
```

```
13935                                   ;;*****************************************************************
13936                                   ;*TEST 720      BUT SERVICE TEST - (JMP @A)
13937                                   ;;*****************************************************************
13938  052700                           TST720:
13939  052700  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13940  052702  012700  000720                   MOV     #720,R0         ;;LOAD R0 WITH TEST NUMBER
13941  052706  013701  052734                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13942  052712  012737  052740  063312           MOV     #3$,@#MBUF0     ;SET UP POINTER--DEST ADDR = 3$ FOR JMP
13943  052720  012746  000020                   MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13944  052724  012746  052734                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13945  052730  000257                           CCC                     ;SCOPE SYNC
13946  052732  000006                           RTT                     ;SET 'T' BIT - GO TO 2$
13947
13948  052734  000177  010352           2$:     JMP     @MBUF0          ;JMP INSTRUCTION SHOULD SPRING TRAP
13949
13950  052740  104005                   3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13951
13952                                   ;;*****************************************************************
13953                                   ;*TEST 721      BUT SERVICE TEST - (RTS PC)
13954                                   ;;*****************************************************************
13955  052742                           TST721:
13956  052742  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13957  052744  012700  000721                   MOV     #721,R0         ;;LOAD R0 WITH TEST NUMBER
13958  052750  013701  053004                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13959  052754  010605                           MOV     SP,R5           ;SAVE THE SP
13960  052756  010737  001010                   MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13961  052762  010506                   1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13962  052764  012746  053006                   MOV     #3$,-(SP)       ;RTS WILL LOAD PC WITH 3$
13963  052770  012746  000020                   MOV     #20,-(SP)       ;SET 'T' BIT IN THE NEW PSW
13964  052774  012746  053004                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13965  053000  000257                           CCC                     ;SCOPE SYNC
13966  053002  000006                           RTT                     ;SET 'T' BIT - GO TO 2$
13967
13968  053004  000207                   2$:     RTS     PC              ;RTS INSTRUCTION SHOULD SPRING TRAP
13969
13970  053006  104005                   3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13971
13972                                   ;;*****************************************************************
13973                                   ;*TEST 722      ALU ADD FUNCTION TEST
13974                                   ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13975                                   ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13976                                   ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13977
13978                                   ;       AIN     BIN     CIN
13979
13980                                   ;       0       0       0
13981                                   ;       0       0       1
13982                                   ;       0       1       0
13983                                   ;       0       1       1
13984                                   ;       1       0       0
13985                                   ;       1       0       1
13986                                   ;       1       1       0
13987                                   ;       1       1       1
13988
13989                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
13990                                   ;TAGGED ''ALUADD'' AS SHOWN BELOW:
```

```
                                        ;ALUADD:        NULL
                                        ;               SRC OP1
                                        ;               DST OP1
                                        ;               SUM1
                                        ;               SRC OP2
                                        ;               DST OP2
                                        ;               SUM2
                                        ;               ETC.

                                        ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
                                        ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
                                        ;;*****************************************************************
053010                                  TST722:
053010  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
053012  012700  000722                      MOV     #722,R0         ;;LOAD R0 WITH TEST NUMBER
053016  012705  063340                1$:   MOV     #ALUADD+4,R5    ;R5 POINTS TO TABLE OF NO.S
053022  010737  001010                      MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
053026  024545                            CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
                                                                    ;(OR NULL ENTRY FIRST TIME THROUGH)
053030  005725                        4$: TST     (R5)+           ;POINT TO A SRC OP
053032  022705  063416                    CMP     #ALUADD+62,R5   ;DONE ALL NO.S IN TABLE ?
053036  001413                            BEQ     5$              ;BR IF YES
053040  012501                            MOV     (R5)+,R1        ;LOAD SRC OP
053042  012503                            MOV     (R5)+,R3        ;LOAD DEST OP
053044  000257                            CCC                     ;SCOPE SYNC

053046  060103                        2$: ADD     R1,R3           ;TEST THE ADD FUNCTION

053050  021503                            CMP     (R5),R3         ;CORRECT SUM ?
053052  001766                            BEQ     4$              ;GO ADD NEXT PAIR IF YES

053054  011504                            MOV     (R5),R4         ;GET S / B SUM
053056  014502                            MOV     -(R5),R2        ;GET DEST OP
053060  104010                        3$: ERROR   10              ;ALU ADD OPERATION FAILED

053062  005725                            TST     (R5)+           ;CORRECT R5 POINTER
053064  000761                            BR      4$              ;GO DO NEXT PAIR

053066  012737  053016  001010        5$: MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR

                                        ;;*****************************************************************
                                        ;*TEST 723      ALU SUB FUNCTION TEST
                                        ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
                                        ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
                                        ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:

                                        ;       AIN     BIN     CIN

                                        ;       0       0       0
                                        ;       0       0       1
                                        ;       0       1       0
                                        ;       0       1       1
                                        ;       1       0       0
                                        ;       1       0       1
                                        ;       1       1       0
```

```
14047                                   ;       1       1       1
14048
14049                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
14050                                   ;TAGGED ''ALUADD'' AS SHOWN BELOW:
14051
14052                                           ;ALUSUB:        NULL
14053                                   ;                       SRC OP1
14054                                   ;                       DST OP1
14055                                   ;                       DIFF1
14056                                   ;                       SRC OP2
14057                                   ;                       DST OP2
14058                                   ;                       DIFF2
14059                                   ;                       ETC.
14060
14061                                   ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14062                                   ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14063                                   ;;*****************************************************************
14064   053074                          TST723:
14065   053074  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14066   053076  012700  000723                 MOV     #723,R0         ;;LOAD R0 WITH TEST NUMBER
14067   053102  012705  063560         1$:     MOV     #ALUSUB+4,R5    ;R5 POINTS TO TABLE OF NO.S
14068   053106  010737  001010                 MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14069   053112  024545                         CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14070                                                                   ;(OR NULL ENTRY FIRST TIME THROUGH)
14071   053114  005725                 4$:     TST     (R5)+           ;POINT TO A SRC OP
14072   053116  022705  063636                 CMP     #ALUSUB+62,R5   ;DONE ALL NO.S IN TABLE ?
14073   053122  001413                         BEQ     5$              ;BR IF YES
14074   053124  012501                         MOV     (R5)+,R1        ;LOAD SRC OP
14075   053126  012503                         MOV     (R5)+,R3        ;LOAD DEST OP
14076   053130  000257                         CCC                     ;SCOPE SYNC
14077
14078   053132  160103                 2$:     SUB     R1,R3           ;TEST THE SUB FUNCTION
14079
14080   053134  021503                         CMP     (R5),R3         ;CORRECT DIFF. ?
14081   053136  001766                         BEQ     4$              ;GO SUB NEXT PAIR IF YES
14082
14083   053140  011504                         MOV     (R5),R4         ;GET S / B DIFF
14084   053142  014502                         MOV     -(R5),R2        ;GET DEST OP
14085   053144  104010                 3$:     ERROR   10              ;ALU SUB OPERATION FAILED
14086
14087   053146  005725                         TST     (R5)+           ;CORRECT R5 POINTER
14088   053150  000761                         BR      4$              ;GO DO NEXT PAIR
14089
14090   053152  012737  053102  001010 5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
14091
14092                                   ;;*****************************************************************
14093                                   ;*TEST 724       ALU ''AND'' FUNCTION TEST USING BIC INSTRUCTION
14094                                   ;THIS TEST VERIFIES THAT THE ALU ''AND'' FUNCTION RESPONDS CORRECTLY
14095                                   ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
14096                                   ;IT  EXECUTES THE BIC INSTRUCTION FOR THE FOLLOWING PAIRS OF
14097                                   ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
14098
14099                                           ;SOURCE OP      DEST. OP        RESULT
14100
14101                                           ;000000         000000          000000
14102                                           ;177777         177777          000000
```

```
                        ;000000         177777          177777
                        ;177777         000000          000000
                        ;125252         125252          000000
                        ;052525         052525          000000
                        ;125252         052525          052525
                        ;052525         125252          125252

                ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TEBLE TAGGED
                ;''ANDTAB'' IN THE FOLLOWING PATTERN:
                        ;ANDTAB:        NULL
                ;                       SRC OP1
                ;                       DST OP1
                ;                       ANS1
                ;                       SRC OP2
                ;                       DST OP2
                ;                       ANS2
                ;                       ETC.

                ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
                ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
                ;;*****************************************************************
053160          TST724:
                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
000004                  MOV     #724,R0                 ;;LOAD R0 WITH TEST NUMBER
012700  000724  1$:     MOV     #ANDTAB+4,R5            ;R5 POINTS TO TABLE OF TEST NO.S
012705  063420          MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
010737  001010          CMP     -(R5),-(R5)             ;RESET R5 TO POINT TO BAD GUYS
024545                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
005725          4$:     TST     (R5)+                   ;POINT TO A SOURCE OPR
022705  063476          CMP     #ANDTAB+62,R5           ;DONE ALL COMBINATIONS ?
001413                  BEQ     5$                      ;BR IF YES
012501                  MOV     (R5)+,R1                ;LOAD THE SRC OP
012503                  MOV     (R5)+,R3                ;LOAD THE DEST OP
000257                  CCC                             ;SCOPE SYNC

040103          2$:     BIC     R1,R3                   ;TEST THE ''AND''

020315                  CMP     R3,(R5)                 ;RESULT CORRECT ?
001766                  BEQ     4$                      ;BR IF YES - GET THE NEXT PAIR

011504                  MOV     (R5),R4                 ;GET THE S / B DATA
014502                  MOV     -(R5),R2                ;GET DEST OP
104010          3$:     ERROR   10                      ;ALU ''AND'' FAILED

005725                  TST     (R5)+                   ;CORRECT R5 POINTER
000761                  BR      4$                      ;GO GET NEXT PAIR

012737  053166  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR

                ;;*****************************************************************
                ;*TEST 725       ALU ''OR'' FUNCTION TEST USING BIS INSTRUCTION
                ;THIS TEST VERIFIES THAT THE ALU ''OR'' FUNCTION RESPONDS CORRECTLY
                ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
                ;IT  EXECUTES THE BIS INSTRUCTION FOR THE FOLLOWING PAIRS OF
                ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
```

```
14159
14160
14161
14162
14163
14164
14165
14166
14167
14168
14169
14170
14171
14172
14173
14174
14175
14176
14177
14178
14179
14180
14181
14182
14183
14184   053244
14185   053244   000004
14186   053246   012700   000725
14187   053252   012705   063500
14188   053256   010737   001010
14189   053262   024545
14190
14191   053264   005725
14192   053266   022705   063556
14193   053272   001413
14194   053274   012501
14195   053276   012503
14196   053300   000257
14197
14198   053302   050103
14199
14200   053304   020315
14201   053306   001766
14202
14203   053310   011504
14204   053312   014502
14205   053314   104010
14206
14207   053316   005725
14208   053320   000761
14209
14210   053322   012737   053252   001010
14211
14212
14213
14214
```

```
                ;SOURCE OP      DEST. OP        RESULT

                ;000000         000000          000000
                ;177777         177777          177777
                ;000000         177777          177777
                ;177777         000000          177777
                ;125252         125252          125252
                ;052525         052525          052525
                ;125252         052525          177777
                ;052525         125252          177777

;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TABLE TAGGED
;''ORTAB'' IN THE FOLLOWING PATTERN:
                ;ORTAB: NULL
;                       SRC OP1
;                       DST OP1
;                       ANS1
;                       SRC OP2
;                       DST OP2
;                       ANS2
;                       ETC.

;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
;;*****************************************************************
TST725:
                SCOPE                           ;CALL THE SCOPE LOOP UTILITY
                MOV     #725,R0                 ;;LOAD R0 WITH TEST NUMBER
1$:             MOV     #ORTAB+4,R5             ;R5 POINTS TO TABLE OF TEST NO.S
                MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
                CMP     -(R5),-(R5)             ;RESET R5 TO POINT TO BAD GUYS
                                                ;(OR NULL ENTRY FIRST TIME THROUGH)
4$:             TST     (R5)+                   ;POINT TO A SOURCE OPR
                CMP     #ORTAB+62,R5            ;DONE ALL COMBINATIONS ?
                BEQ     5$                      ;BR IF YES
                MOV     (R5)+,R1                ;LOAD THE SRC OP
                MOV     (R5)+,R3                ;LOAD THE DEST OP
                CCC                             ;SCOPE SYNC

2$:             BIS     R1,R3                   ;TEST THE ''OR''

                CMP     R3,(R5)                 ;RESULT CORRECT ?
                BEQ     4$                      ;BR IF YES - GET THE NEXT PAIR

                MOV     (R5),R4                 ;GET THE S / B DATA
                MOV     -(R5),R2                ;GET DEST OP
3$:             ERROR   10                      ;ALU ''OR'' FAILED

                TST     (R5)+                   ;CORRECT R5 POINTER
                BR      4$                      ;GO GET NEXT PAIR

5$:             MOV     #1$,@#$LPERR            ;LOOP FROM BEGINNING ON ERROR

;;*****************************************************************
;*TEST 726       INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
```

```
14215
14216
14217                                   ;TEST SEQUENCE:
14218                                   ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14219                                   ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14220                                   ;3. THE SOURCE OP IS INCREMENTED
14221                                   ;4. THE DEST OP IS DECREMENTED
14222                                   ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE SOURCE OP GOES
14223                                   ;   NEGATIVE
14224
14225                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14226
14227                                   ;       1. SW09=0  THE TEST IS EXITED
14228                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14229
14230  053330                           ;;*****************************************************************
14231  053330  000004                   TST726:
14232  053332  012700  000726                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14233  053336  005001                           MOV     #726,R0         ;;LOAD R0 WITH TEST NUMBER
14234  053340  005002           10$:    CLR     R1                      ;INITIALIZE REGS TO 000000
14235  053342  005004                   CLR     R2
14236  053344  010737  001010           CLR     R4
14237  053350  010203                   MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
14238  053352  000257           1$:     MOV     R2,R3                   ;LOAD DEST OPERAND
14239                                   CCC                             ;SCOPE SYNC
14240  053354  060103           2$:     ADD     R1,R3                   ;ADD THE TWO TEST NO.S
14241                                                                   ;RESULT S / B = 000000
14242
14243  053356  020403                   CMP     R4,R3                   ;RESULT = 000000 ?
14244  053360  001402                   BEQ     4$                      ;BR IF YES
14245
14246  053362  104010           3$:     ERROR   10                      ;INCORRECT RESULT IN R3
14247
14248  053364  000407                   BR      TST727                  ;;EXIT TO NEXT TEST
14249
14250  053366  005201           4$:     INC     R1.                     ;ADD 1 TO SOURCE OP
14251  053370  100402                   BMI     5$                      ;GET OUT IF IT WENT NEGATIVE
14252  053372  005302                   DEC     R2                      ;SUB 1 FROM THE DEST OP
14253  053374  000765                   BR      1$                      ;GO ADD THE TWO NO.S
14254
14255  053376  012737  053336  001010   5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14256
14257                                   ;;*****************************************************************
14258                                   ;*TEST 727      INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
14259                                   ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
14260                                   ;TEST SEQUENCE:
14261
14262                                   ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14263                                   ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14264                                   ;3. THE SOURCE OP IS DECREMENTED
14265                                   ;4. THE DEST OP IS INCREMENTED
14266                                   ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE DEST. OP GOES
14267                                   ;   NEGATIVE
14268
14269                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14270
```

```
14271                                   ;       1. SW09=0  THE TEST IS EXITED
14272                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14273                                   ;;*****************************************************************
14274  053404                           TST727:
14275  053404  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14276  053406  012700  000727                   MOV     #727,R0         ;;LOAD R0 WITH TEST NUMBER
14277  053412  005001                   10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14278  053414  005002                           CLR     R2
14279  053416  005004                           CLR     R4
14280  053420  010737  001010                   MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14281  053424  010203                   1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14282  053426  000257                           CCC                     ;SCOPE SYNC
14283
14284  053430  060103                   2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14285                                           ;RESULT S / B = 000000
14286
14287  053432  020403                           CMP     R4,R3           ;RESULT = 000000 ?
14288  053434  001402                           BEQ     4$              ;BR IF YES
14289
14290  053436  104010                   3$:     ERROR   10              ;INCORRECT RESULT IN R3
14291
14292  053440  000407                           BR      TST730          ;;GO TO SCOPE EXIT
14293
14294  053442  005202                   4$:     INC     R2              ;ADD 1 TO DEST. OP
14295  053444  100402                           BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14296  053446  005301                           DEC     R1              ;SUB 1 FROM THE SOURCE OP
14297  053450  000765                           BR      1$              ;GO ADD THE TWO NO.S
14298
14299  053452  012737  053412  001010   5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14300
14301                                   ;;*****************************************************************
14302                                   ;*TEST 730       MUL RA,RB TEST ; N:C = 1111
14303                                   ;;*****************************************************************
14304  053460                           TST730:
14305  053460  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14306  053462  012700  000730                   MOV     #730,R0         ;;LOAD R0 WITH TEST NUMBER
14307  053466  013737  053516  001076           MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14308  053474  005001                           CLR     R1              ;S/B RESULT IN R2
14309  053476  012704  000006                   MOV     #6,R4           ;S/B RESULT IN R3
14310  053502  012702  000002                   MOV     #2,R2           ;INITIALIZE REG
14311  053506  005003                           CLR     R3              ;INITIALIZE REG + 1
14312  053510  012705  000003                   MOV     #3,R5           ;INITIALIZE SRC
14313  053514  000277                           SCC                     ;SCOPE SYNC
14314
14315  053516  070205                   2$:     MUL     R5,R2           ;TEST THE MUL
14316
14317  053520  100403                           BMI     3$              ;N:C=0000?
14318  053522  001402                           BEQ     3$
14319  053524  102401                           BVS     3$
14320  053526  103001                           BCC     4$
14321
14322  053530  104044                   3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14323
14324  053532  020304                   4$:     CMP     R3,R4           ;REG+1 CORRECT?
14325  053534  001002                           BNE     5$              ;BR IF NOT
14326  053536  020102                           CMP     R1,R2           ;REG CORRECT?
```

```
14327  053540  001401                    BEQ     TST731          ;;BR IF YES
14328
14329  053542  104045                5$: ERROR   45              ;MUL DELIVERED WRONG RESULT
14330
14331                                  ;;*****************************************************************
14332                                  ;*TEST 731      MUL (RA),RB TEST ; N:C = 0000-SET C
14333                                  ;;*****************************************************************
14334  053544                          TST731:
14335  053544  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14336  053546  012700  000731                MOV     #731,R0         ;;LOAD R0 WITH TEST NUMBER
14337  053552  013737  053606  001076        MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14338  053560  005001                        CLR     R1              ;S/B RESULT IN R2
14339  053562  012704  123450                MOV     #123450,R4      ;S/B RESULT IN R3
14340  053566  012702  012345                MOV     #012345,R2      ;INITIALIZE REG
14341  053572  005003                        CLR     R3              ;INITIALIZE REG + 1
14342  053574  012705  063312                MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14343  053600  012715  000010                MOV     #10,(R5)        ;INITIALIZE SRC
14344  053604  000257                        CCC                     ;SCOPE SYNC
14345
14346  053606  070215                2$: MUL     (R5),R2             ;TEST THE MUL
14347
14348  053610  100403                        BMI     3$              ;N:C=0001?
14349  053612  001402                        BEQ     3$
14350  053614  102401                        BVS     3$
14351  053616  103401                        BCS     4$
14352
14353  053620  104044                3$: ERROR   44              ;COND CODES SET IMPROPERLY
14354
14355  053622  020304                4$: CMP     R3,R4           ;REG+1 CORRECT?
14356  053624  001002                        BNE     5$              ;BR IF NOT
14357  053626  020102                        CMP     R1,R2           ;REG CORRECT?
14358  053630  001401                        BEQ     TST732          ;;BR IF YES
14359
14360  053632  104045                5$: ERROR   45              ;MUL DELIVERED WRONG RESULT
14361
14362                                  ;;*****************************************************************
14363                                  ;*TEST 732      MUL (RA)+,RB TEST ; N:C = 0000-SET Z
14364                                  ;;*****************************************************************
14365  053634                          TST732:
14366  053634  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14367  053636  012700  000732                MOV     #732,R0         ;;LOAD R0 WITH TEST NUMBER
14368  053642  013737  053674  001076        MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14369  053650  005001                        CLR     R1              ;S/B RESULT IN R2
14370  053652  005004                        CLR     R4              ;S/B RESULT IN R3
14371  053654  005002                        CLR     R2              ;INITIALIZE REG
14372  053656  012703  177777                MOV     #-1,R3          ;INITIALIZE REG + 1
14373  053662  012705  063312                MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14374  053666  012715  000010                MOV     #10,(R5)        ;INITIALIZE SRC
14375  053672  000257                        CCC                     ;SCOPE SYNC
14376
14377  053674  070225                2$: MUL     (R5)+,R2            ;TEST THE MUL
14378
14379  053676  100403                        BMI     3$              ;N:C=0100?
14380  053700  001002                        BNE     3$
14381  053702  102401                        BVS     3$
14382  053704  103001                        BCC     4$
```

```
14383
14384  053706  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14385
14386  053710  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14387  053712  001002                          BNE     5$              ;BR IF NOT
14388  053714  020102                          CMP     R1,R2           ;REG CORRECT?
14389  053716  001401                          BEQ     6$              ;BR IF YES
14390
14391  053720  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14392
14393  053722  022705  063314          6$:     CMP     #MBUF0+2,R5     ;DID R5 GET AUTO-INCREMENTED?
14394  053726  001401                          BEQ     TST733          ;;BR IF YES
14395
14396  053730  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14397
14398                                  ;;*********************************************************************
14399                                  ;*TEST 733      MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST = -,+
14400                                  ;;*********************************************************************
14401  053732                          TST733:
14402  053732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14403  053734  012700  000733                  MOV     #733,R0         ;;LOAD R0 WITH TEST NUMBER
14404  053740  013737  054000  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14405  053746  012701  177777                  MOV     #-1,R1          ;S/B RESULT IN R2
14406  053752  012704  177770                  MOV     #-10,R4         ;S/B RESULT IN R3
14407  053756  012702  000001                  MOV     #1,R2           ;INITIALIZE REG
14408  053762  005003                          CLR     R3              ;INITIALIZE REG + 1
14409  053764  012705  063306                  MOV     #ATA+10,R5      ;SET UP POINTER TO POINTER TO MBUF0
14410  053770  012737  177770  063312          MOV     #-10,@#MBUF0    ;INITIALIZE SRC
14411  053776  000257                          CCC                     ;SCOPE SYNC
14412
14413  054000  070235                  2$:     MUL     @(R5)+,R2       ;TEST THE MUL
14414
14415  054002  100003                          BPL     3$              ;N:C=1000?
14416  054004  001402                          BEQ     3$
14417  054006  102401                          BVS     3$
14418  054010  103001                          BCC     4$
14419
14420  054012  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14421
14422  054014  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14423  054016  001002                          BNE     5$              ;BR IF NOT
14424  054020  020102                          CMP     R1,R2           ;REG CORRECT?
14425  054022  001401                          BEQ     6$              ;BR IF YES
14426
14427  054024  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14428
14429  054026  022705  063310          6$:     CMP     #ATA+12,R5      ;DID R5 GET AUTO-INCREMENTED?
14430  054032  001401                          BEQ     TST734          ;;BR IF YES
14431
14432  054034  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14433
14434                                  ;;*********************************************************************
14435                                  ;*TEST 734      MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK = +,-
14436                                  ;;*********************************************************************
14437  054036                          TST734:
14438  054036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
054040  012700  000734                  MOV     #734,R0          ;;LOAD R0 WITH TEST NUMBER
054044  013737  054104  001076          MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
054052  012701  177777                  MOV     #-1,R1           ;S/B RESULT IN R2
054056  012704  177770                  MOV     #-10,R4          ;S/B RESULT IN R3
054062  012702  177777                  MOV     #-1,R2           ;INITIALIZE REG
054066  005003                          CLR     R3               ;INITIALIZE REG + 1
054070  012705  063314                  MOV     #MBUF0+2,R5      ;SET UP POINTER TO SRC
054074  012737  000010  063312          MOV     #10,@#MBUF0      ;INITIALIZE SRC
054102  000277                          SCC                      ;SCOPE SYNC

054104  070245                  2$:     MUL     -(R5),R2         ;TEST THE MUL

054106  100003                          BPL     3$               ;N:C=1000?
054110  001402                          BEQ     3$
054112  102401                          BVS     3$
054114  103001                          BCC     4$

054116  104044                  3$:     ERROR   44               ;COND CODES SET IMPROPERLY

054120  020304                  4$:     CMP     R3,R4            ;REG+1 CORRECT?
054122  001002                          BNE     5$               ;BR IF NOT
054124  020102                          CMP     R1,R2            ;REG CORRECT?
054126  001401                          BEQ     6$               ;BR IF YES

054130  104045                  5$:     ERROR   45               ;MUL DELIVERED WRONG RESULT

054132  022705  063312          6$:     CMP     #MBUF0,R5        ;DID SRC REG GET AUTO-DECREMENTED?
054136  001401                          BEQ     TST735           ;;BR IF YES

054140  104046                          ERROR   46               ;AUTO DECREMENT DID NOT OCCUR

                                ;;*******************************************************************
                                ;*TEST 735       MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
                                ;;*******************************************************************
054142                          TST735:
054142  000004                          SCOPE                    ;CALL THE SCOPE LOOP UTILITY
054144  012700  000735                  MOV     #735,R0          ;;LOAD R0 WITH TEST NUMBER
054150  013737  054210  001076          MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
054156  005001                          CLR     R1               ;S/B RESULT IN R2
054160  012704  106420                  MOV     #106420,R4       ;/S/B RESULT IN R3
054164  012702  177776                  MOV     #-2,R2           ;INITIALIZE REG
054170  012703  177777                  MOV     #-1,R3           ;INITIALIZE REG + 1
054174  012705  063310                  MOV     #ATA+12,R5       ;SET UP POINTER TO POINTER TO MBUF0
054200  012737  134570  063312          MOV     #-43210,@#MBUF0  ;INITIALIZE SRC
054206  000277                          SCC                      ;SCOPE SYNC

054210  070255                  2$:     MUL     @-(R5),R2        ;TEST THE MUL

054212  100403                          BMI     3$               ;N:C=0001?
054214  001402                          BEQ     3$
054216  102401                          BVS     3$
054220  103401                          BCS     4$

054222  104044                  3$:     ERROR   44               ;COND CODES SET IMPROPERLY

054224  020304                  4$:     CMP     R3,R4            ;REG+1 CORRECT?
```

```
14495  054226  001002                    BNE     5$              ;BR IF NOT
14496  054230  020102                    CMP     R1,R2           ;REG CORRECT?
14497  054232  001401                    BEQ     6$              ;BR IF YES
14498
14499  054234  104045            5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14500
14501  054236  022705  063306    6$:     CMP     #ATA+10,R5      ;DID R5 GET AUTO-DECREMENTED?
14502  054242  001401                    BEQ     TST736          ;;BR IF YES
14503
14504  054244  104046                    ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14505
14506                            ;;**********************************************************************
14507                            ;*TEST 736      MUL X(RA),RB TEST ; N:C = 1111 TO 0100
14508                            ;;**********************************************************************
14509  054246                    TST736:
14510  054246  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14511  054250  012700  000736            MOV     #736,R0         ;;LOAD R0 WITH TEST NUMBER
14512  054254  013737  054310  001076    MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14513  054262  005001                    CLR     R1              ;S/B RESULT IN R2
14514  054264  005004                    CLR     R4              ;S/B RESULT IN R3
14515  054266  012702  012345            MOV     #012345,R2      ;INITIALIZE REG
14516  054272  012703  177777            MOV     #-1,R3          ;INITIALIZE REG + 1
14517  054276  012705  063312            MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14518  054302  005065  000002            CLR     2(R5)           ;INITIALIZE SRC
14519  054306  000277                    SCC                     ;SCOPE SYNC
14520
14521  054310  070265  000002    2$:     MUL     2(R5),R2        ;TEST THE MUL
14522
14523  054314  100403                    BMI     3$              ;N:C=0100?
14524  054316  001002                    BNE     3$
14525  054320  102401                    BVS     3$
14526  054322  103001                    BCC     4$
14527
14528  054324  104044            3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14529
14530  054326  020304            4$:     CMP     R3,R4           ;REG+1 CORRECT?
14531  054330  001002                    BNE     5$              ;BR IF NOT
14532  054332  020102                    CMP     R1,R2           ;REG CORRECT?
14533  054334  001401                    BEQ     TST737          ;;BR IF YES
14534
14535  054336  104045            5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14536
14537                            ;;**********************************************************************
14538                            ;*TEST 737      MUL @X(RA),RB TEST
14539                            ;;**********************************************************************
14540  054340                    TST737:
14541  054340  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14542  054342  012700  000737            MOV     #737,R0         ;;LOAD R0 WITH TEST NUMBER
14543  054346  013737  054404  001076    MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14544  054354  005001                    CLR     R1              ;S/B RESULT IN R2
14545  054356  012704  000100            MOV     #100,R4         ;S/B RESULT IN R3
14546  054362  012702  000010            MOV     #10,R2          ;INITIALIZE REG
14547  054366  005003                    CLR     R3              ;INITIALIZE REG + 1
14548  054370  012705  063276            MOV     #ATA,R5         ;GET POINTER TO TABLE OF POINTERS
14549  054374  012737  000010  063312    MOV     #10,@#MBUF0     ;INITIALIZE SRC
14550  054402  000257                    CCC                     ;SCOPE SYNC
```

```
14551
14552  054404  070275  000010          2$:     MUL     @10(R5),R2      ;TEST THE MUL
14553
14554  054410  020304                          CMP     R3,R4           ;REG+1 CORRECT?
14555  054412  001002                          BNE     3$              ;BR IF NOT
14556  054414  020102                          CMP     R1,R2           ;REG CORRECT?
14557  054416  001401                          BEQ     TST740          ;;BR IF YES
14558
14559  054420  104045                  3$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14560
14561                                  ;;*********************************************************************
14562                                  ;*TEST 740       DIV #N,RA TEST ; N:C = 1111
14563                                  ;;*********************************************************************
14564  054422                          TST740:
14565  054422  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14566  054424  012700  000740                  MOV     #740,R0         ;;LOAD R0 WITH TEST NUMBER
14567  054430  013737  054456  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14568  054436  012701  010000                  MOV     #010000,R1      ;S/B RES IN R2
14569  054442  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14570  054446  005002                          CLR     R2              ;SET UP REG OPERAND
14571  054450  012703  020001                  MOV     #020001,R3      ;SET UP REG+1 OP
14572  054454  000277                          SCC                     ;SCOPE SYNC
14573
14574  054456  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14575
14576  054462  100403                          BMI     3$              ;N:C=0000?
14577  054464  001402                          BEQ     3$
14578  054466  102401                          BVS     3$
14579  054470  103001                          BCC     4$
14580
14581  054472  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14582
14583  054474  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14584  054476  001002                          BNE     5$              ;BR IF NOT
14585  054500  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14586  054502  001401                          BEQ     TST741          ;;BR IF YES
14587
14588  054504  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14589
14590                                  ;;*********************************************************************
14591                                  ;*TEST 741       DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
14592                                  ;;*********************************************************************
14593  054506                          TST741:
14594  054506  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14595  054510  012700  000741                  MOV     #741,R0         ;;LOAD R0 WITH TEST NUMBER
14596  054514  013737  054544  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14597  054522  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14598  054526  012704  177776                  MOV     #-2,R4          ;S/B RES IN R3
14599  054532  012702  177777                  MOV     #-1,R2          ;SET UP REG OPERAND
14600  054536  012703  177762                  MOV     #-14.,R3        ;SET UP REG+1 OP
14601  054542  000257                          CCC                     ;SCOPE SYNC
14602
14603  054544  071227  000004          2$:     DIV     #4,R2           ;TEST DIV
14604
14605  054550  100003                          BPL     3$              ;N:C=1000?
14606  054552  001402                          BEQ     3$
```

```
14607  054554  102401                          BVS     3$
14608  054556  103001                          BCC     4$
14609
14610  054560  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14611
14612  054562  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14613  054564  001002                          BNE     5$              ;BR IF NOT
14614  054566  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14615  054570  001401                          BEQ     TST742          ;;BR IF YES
14616
14617  054572  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14618
14619                                  ;;*********************************************************************
14620                                  ;*TEST 742       DIV #N,RA TEST ; N:C = 0000 TO 0100
14621                                  ;;*********************************************************************
14622  054574                          TST742:
14623  054574  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14624  054576  012700  000742                  MOV     #742,R0         ;;LOAD R0 WITH TEST NUMBER
14625  054602  013737  054626  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14626  054610  005001                          CLR     R1              ;S/B RES IN R2
14627  054612  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14628  054616  005002                          CLR     R2              ;SET UP REG OPERAND
14629  054620  012703  000001                  MOV     #1,R3           ;SET UP REG+1 OP
14630  054624  000257                          CCC                     ;SCOPE SYNC
14631
14632  054626  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14633
14634  054632  100403                          BMI     3$              ;N:C=0100?
14635  054634  001002                          BNE     3$
14636  054636  102401                          BVS     3$
14637  054640  103001                          BCC     4$
14638
14639  054642  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14640
14641  054644  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14642  054646  001002                          BNE     5$              ;BR IF NOT
14643  054650  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14644  054652  001401                          BEQ     TST743          ;;BR IF YES
14645
14646  054654  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14647
14648                                  ;;*********************************************************************
14649                                  ;*TEST 743       DIV #-N,RA TEST ; RA POS
14650                                  ;;*********************************************************************
14651  054656                          TST743:
14652  054656  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14653  054660  012700  000743                  MOV     #743,R0         ;;LOAD R0 WITH TEST NUMBER
14654  054664  013737  054712  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14655  054672  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14656  054676  012704  000002                  MOV     #2,R4           ;S/B RES IN R3
14657  054702  005002                          CLR     R2              ;SET UP REG OPERAND
14658  054704  012703  000016                  MOV     #14.,R3         ;SET UP REG+1 OP
14659  054710  000257                          CCC                     ;SCOPE SYNC
14660
14661  054712  071227  177774          2$:     DIV     #-4,R2          ;TEST DIV
14662
```

```
14663  054716  020304                  CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14664  054720  001002                  BNE     3$              ;BR IF NOT
14665  054722  020102                  CMP     R1,R2           ;CORRECT RESULT IN REG?
14666  054724  001401                  BEQ     TST744          ;;BR IF YES
14667
14668  054726  104045          3$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14669
14670                          ;;**********************************************************************
14671                          ;*TEST 744      DIV  TEST - V BIT GETS SET
14672                          ;*      THIS TEST TESTS THAT THE V BIT CAN BE SET IN ALL THE
14673                          ;*      POSSIBLE WAYS.  SINCE THE INSTRUCTION SHOULD BE ABORTED, THE
14674                          ;*      RESULTS CANNOT BE GUARANTEED.  FOR THIS REASON, ONLY
14675                          ;*      THE CONDITION CODES ARE CHECKED.
14676                          ;;**********************************************************************
14677  054730                  TST744:
14678  054730  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14679  054732  012700  000744          MOV     #744,R0         ;;LOAD R0 WITH TEST NUMBER
14680  054736  013701  054756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14681  054742  012704  000002          MOV     #2,R4           ;S/B PSW
14682  054746  005037  177776          CLR     @#PSW           ;CLEAR OUT OTHER PSW BITS
14683  054752  012702  000050          MOV     #50,R2          ;SET UP REG OP
14684
14685  054756  071227  000005  2$:     DIV     #5,R2           ;TEST DIV -- SHOULD ABORT
14686
14687  054762  100424                  BMI     3$              ;N:C=0010?
14688  054764  001423                  BEQ     3$
14689  054766  102022                  BVC     3$
14690  054770  103421                  BCS     3$
14691
14692  054772  012702  177777          MOV     #-1,R2          ;INITIALIZE REG OP
14693  054776  005003                  CLR     R3              ;INITILAIZE REG+1 OP
14694
14695  055000  071227  177776          DIV     #-2,R2          ;TEST DIV -- SHOULD ABORT
14696
14697  055004  100413                  BMI     3$              ;N:C=0010?
14698  055006  001412                  BEQ     3$
14699  055010  102011                  BVC     3$
14700  055012  103410                  BCS     3$
14701
14702  055014  012704  000003          MOV     #3,R4           ;S/B PSW
14703
14704  055020  071227  000000          DIV     #0,R2           ;TEST DIV BY 0 -- SHOULD ABORT
14705
14706  055024  100403                  BMI     3$              ;N:C=0010?
14707  055026  001402                  BEQ     3$
14708  055030  102001                  BVC     3$
14709  055032  103405                  BCS     TST745          ;;IF ALL OK, THEN EXIT TEST
14710
14711  055034  013703  177776  3$:     MOV     @#PSW,R3        ;GET WAS PSW
14712  055040  012702  177776          MOV     #PSW,R2         ;DESTINATION IS PSW
14713
14714  055044  104001                  ERROR   1               ;CONDITION CODES SET WRONG
14715
14716                          ;;**********************************************************************
14717                          ;*TEST 745      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14718                          ;;**********************************************************************
```

```
14719  055046
14720  055046  000004
14721  055050  012700  000745      TST745:
14722  055054  013701  055072              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14723  055060  012704  123450              MOV     #745,R0         ;;LOAD R0 WITH TEST NUMBER
14724  055064  012703  112345              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14725  055070  000257                      MOV     #123450,R4      ;S/B RESULT
14726                                       MOV     #112345,R3      ;INITIAL REG
14727  055072  072327  000003              CCC                     ;SCOPE SYNC
14728                              2$:     ASH     #3,R3           ;TEST THE ASH
14729  055076  100003
14730  055100  001402                      BPL     3$              ;N:C=1010?
14731  055102  102001                      BEQ     3$
14732  055104  103001                      BVC     3$
14733                                       BCC     4$
14734  055106  104002              3$:     ERROR   2               ;INCORRECT CONDITION CODES
14735
14736  055110  020304              4$:     CMP     R3,R4           ;CORRECT RESULT?
14737  055112  001401                      BEQ     TST746          ;;BR IF YES
14738  055114  104002                      ERROR   2               ;ASH DELIVERED WRONG RESULT
14739
14740                              ;;**********************************************************************
14741                              ;*TEST 746       ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14742                              ;;**********************************************************************
14743  055116                      TST746:
14744  055116  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14745  055120  012700  000746              MOV     #746,R0         ;;LOAD R0 WITH TEST NUMBER
14746  055124  013701  055142              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14747  055130  005004                      CLR     R4              ;S/B RESULT
14748  055132  012703  000004              MOV     #4,R3           ;INITIAL REG
14749  055136  000257                      CCC                     ;SCOPE SYNC
14750  055140  000270                      SEN                     ;CODES = 1000
14751
14752  055142  072327  177775      2$:     ASH     #-3,R3          ;TEST THE ASH
14753
14754  055146  100403                      BMI     3$              ;N:C=0101?
14755  055150  001002                      BNE     3$
14756  055152  102401                      BVS     3$
14757  055154  103401                      BCS     4$
14758
14759  055156  104002              3$:     ERROR   2               ;INCORRECT CONDITION CODES
14760
14761  055160  020304              4$:     CMP     R3,R4           ;CORRECT RESULT?
14762  055162  001401                      BEQ     TST747          ;;BR IF YES
14763  055164  104002                      ERROR   2               ;ASH DELIVERED WRONG RESULT
14764
14765                              ;;**********************************************************************
14766                              ;*TEST 747       ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14767                              ;;**********************************************************************
14768  055166                      TST747:
14769  055166  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14770  055170  012700  000747              MOV     #747,R0         ;;LOAD R0 WITH TEST NUMBER
14771  055174  013701  055212              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14772  055200  012704  177234              MOV     #177234,R4      ;S/B RESULT
14773  055204  012703  123432              MOV     #123432,R3      ;INITIAL REG
14774  055210  000277                      SCC                     ;SCOPE SYNC
```

```
14775
14776  055212  072327  177772          2$:     ASH     #-6,R3          ;TEST THE ASH
14777
14778  055216  100003                          BPL     3$              ;N:C=1000?
14779  055220  001402                          BEQ     3$
14780  055222  102401                          BVS     3$
14781  055224  103001                          BCC     4$
14782
14783  055226  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14784
14785  055230  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14786  055232  001401                          BEQ     TST750          ;;BR IF YES
14787  055234  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14788
14789                                  ;;**********************************************************************
14790                                  ;*TEST 750       ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14791                                  ;;**********************************************************************
14792  055236                          TST750:
14793  055236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14794  055240  012700  000750                  MOV     #750,R0         ;;LOAD R0 WITH TEST NUMBER
14795  055244  013737  055274  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14796  055252  012701  123456                  MOV     #123456,R1      ;S/B RES IN R2
14797  055256  012704  076530                  MOV     #076530,R4      ;S/B RES IN R3
14798  055262  012702  112345                  MOV     #112345,R2      ;INITIALIZE COMBINED
14799  055266  012703  147653                  MOV     #147653,R3      ;  REGISTERS
14800  055272  000257                          CCC                     ;SCOPE SYNC
14801
14802  055274  073227  000003          2$:     ASHC    #3,R2           ;TEST ASHC
14803
14804  055300  100003                          BPL     3$              ;N:C=1010?
14805  055302  001402                          BEQ     3$
14806  055304  102001                          BVC     3$
14807  055306  103001                          BCC     4$
14808
14809  055310  104044                  3$:     ERROR   44              ;COND CODES WRONG
14810
14811  055312  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14812  055314  001002                          BNE     5$              ;BR IF NOT
14813  055316  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14814  055320  001401                          BEQ     TST751          ;;BR IF YES
14815  055322  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14816
14817                                  ;;**********************************************************************
14818                                  ;*TEST 751       ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14819                                  ;;**********************************************************************
14820  055324                          TST751:
14821  055324  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14822  055326  012700  000751                  MOV     #751,R0         ;;LOAD R0 WITH TEST NUMBER
14823  055332  013737  055356  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14824  055340  005001                          CLR     R1              ;S/B RES IN R2
14825  055342  005004                          CLR     R4              ;S/B RES IN R3
14826  055344  005002                          CLR     R2              ;INITIALIZE COMBINED
14827  055346  012703  000005                  MOV     #5,R3           ;  REGISTERS
14828  055352  000257                          CCC                     ;SCOPE SYNC
14829  055354  000270                          SEN                     ;CODES = 1000
14830
```

```
14831  055356  073227  177775          2$:     ASHC    #-3,R2          ;TEST ASHC
14832
14833  055362  100403                          BMI     3$              ;N:C=0101?
14834  055364  001002                          BNE     3$
14835  055366  102401                          BVS     3$
14836  055370  103401                          BCS     4$
14837
14838  055372  104044                  3$:     ERROR   44              ;COND CODES WRONG
14839
14840  055374  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14841  055376  001002                          BNE     5$              ;BR IF NOT
14842  055400  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14843  055402  001401                          BEQ     TST752          ;;BR IF YES
14844  055404  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14845
14846                                  ;;*****************************************************************
14847                                  ;*TEST 752       ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1111 TO 1000
14848                                  ;;*****************************************************************
14849  055406                          TST752:
14850  055406  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14851  055410  012700  000752                  MOV     #752,R0         ;;LOAD R0 WITH TEST NUMBER
14852                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT15
14853  055414  032737  100000  063234          BIT     #BIT15,@#BPTLOC ;BREAKPOINT HALT SET ??
14854  055422  001401                          BEQ     .+4             ;BR IF NOT
14855  055424  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
14856  055426  013737  055456  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14857  055434  012701  177234                  MOV     #177234,R1      ;S/B RES IN R2
14858  055440  012704  135275                  MOV     #135275,R4      ;S/B RES IN R3
14859  055444  012702  123456                  MOV     #123456,R2      ;INITIALIZE COMBINED
14860  055450  012703  127542                  MOV     #127542,R3      ;  REGISTERS
14861  055454  000257                          CCC                     ;SCOPE SYNC
14862
14863  055456  073227  177772          2$:     ASHC    #-6,R2          ;TEST ASHC
14864
14865  055462  100003                          BPL     3$              ;N:C=1000?
14866  055464  001402                          BEQ     3$
14867  055466  102401                          BVS     3$
14868  055470  103401                          BCS     4$
14869
14870  055472  104044                  3$:     ERROR   44              ;COND CODES WRONG
14871
14872  055474  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14873  055476  001002                          BNE     5$              ;BR IF NOT
14874  055500  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14875  055502  001401                          BEQ     TST753          ;;BR IF YES
14876  055504  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14877
14878                                  ;*
14879                                  ;*      THIS SECTION OF THE MED TESTS EXERCISES CERTAIN SCRATCH
14880                                  ;*      PAD REGISTERS USING MED READS AND WRITES.  THEIR ORIGINAL
14881                                  ;*      CONTENTS ARE RESTORED BUT:
14882                                  ;*
14883                                  ;*      ******************** IMPORTANT NOTE **********************
14884                                  ;*      *                                                        *
14885                                  ;*      *  THE CONSOLE MUST NOT !!! BE USED DURING THESE MED     *
14886                                  ;*      *  TESTS. NO INTERRUPTS OR TRAPS CAN BE ALLOWED EITHER*
```

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

                                        ;;*****************************************************************
                                        ;*TEST 753      CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
                                        ;*      THE NEXT TEST BELOW CHECKS TO SEE THAT THE 'MED''
                                        ;*      (MAINTENANCE, EXAM, AND DEPOSIT) INSTRUCTION WILL EXECUTE
                                        ;*      WHEN IN KERNEL MODE WITHOUT AFFECTING THE PSW AND
                                        ;*      THAT IT IS ILLEGAL IN USER MODE
                                        ;;*****************************************************************
055506                                  TST753:
055506  012700  000752                          MOV     #752,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
055512  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
055514  012737  000304  177770          MED1:   MOV     #304,@#UBREAK   ;SET SCOPE SYNC FOR MED INSTR
055522  012737  140000  177776                  MOV     #140000,@#PSW   ;GO TO USER MODE
055530  012706  001000                          MOV     #STACK,SP       ;SETUP USER STACK PTR.
055534  012737  055566  000004                  MOV     #2$,@#ERRVEC    ;SET ERROR TRAP VECTOR TO 2$ BELOW
055542  012737  055566  000010                  MOV     #2$,@#RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
055550  012701  177777                          MOV     #-1,R1          ;LOAD R1 WITH A -1
055554  005000                                  CLR     R0              ;CLEAR R0
055556  076600                                  MED                     ;TRY TO DO MAINT. EXAMINE
055560  000041                                  .WORD   041             ;MED READ CODE FOR R1
055562  104012                                  ERROR   12              ;ERROR - MED INST. NOT ILLEGAL IN USER
055564  000404                                  BR      4$
055566  005700                          2$:     TST     R0              ;IS R0 UNCHANGED?
055570  001401                                  BEQ     3$              ;BRANCH IF YES
055572  104013                                  ERROR   13              ;ERROR - MED INSTRUCTION WAS EXECUTED
                                                                        ;BEFORE TRAPPING
055574  022626                          3$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
055576  012737  061220  000004          4$:     MOV     #BERR,@#ERRVEC  ;RESTORE ERROR TRAP VECTOR
055604  012737  061122  000010                  MOV     #RSERR,@#RESVEC ;RESTORE RESERVED INST. TRAP VECTOR

055612  005037  177776                  MED0:   CLR     @#PSW           ;GO TO KERNEL MODE,CLEAR COND. CODES
055616  076600                                  MED                     ;DO MAINT. EXAMINE OF R1
055620  000041                                  .WORD   041             ;MED READ CODE FOR R1
055622  103403                                  BCS     MEDHLT
055624  102402                                  BVS     MEDHLT
055626  100401                                  BMI     MEDHLT
055630  001001                                  BNE     .+4
055632  104014                          MEDHLT: ERROR   14              ;ERROR CC-BITS IN PSW AFFECTED BY MED

                                        ;;*****************************************************************
                                        ;*TEST 754      MED TEST - R/W DATA PATTERNS TO REGS
                                        ;*      THIS PARTICULAR MED TEST WRITES DATA PATTERNS
                                        ;*      TO THOSE INTERNAL REGS. WHICH CAN BE WRITTEN
                                        ;*      AND READ WITHOUT SPECIAL CONSIDERAIONS.  REGISTERS
                                        ;*      REQUIRING SPECIAL TESTS ARE TESTED IN LATER
                                        ;*      MED TESTS.
                                        ;*      TABLE II CONTAINS THE REGISTER ADDRESSES.
                                        ;*
                                        ;*      A MAX. OF 3 ERRORS ARE REPORTED FOR EACH LOC.
                                        ;*
                                        ;;*****************************************************************
055634                                  TST754:
055634  012700  000753                          MOV     #753,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
```

```
14943  055640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14944  055642  012737  000340  177776  MEDT1:  MOV     #340,@#PSW      ;KERNEL MODE-PRIORITY 7
14945  055650  012701  064166                  MOV     #TBL2,R1        ;INITIALIZE ADDRESS POINTER
14946  055654  012737  125252  001102  1$:     MOV     #125252,@#$TMP2
14947  055662  111137  055730                  MOVB    (R1),@#11$      ;PUT WRITE CODE BY 'WRITE-MED'S'
14948  055666  112137  055752                  MOVB    (R1)+,@#13$     ;AND POINT R1 TO READ CODE
14949  055672  111137  055710                  MOVB    (R1),@#10$      ;PUT READ CODE BY 'READ-MED'S'
14950  055676  112137  055736                  MOVB    (R1)+,@#12$     ;R1 NOW POINTS TO NEXT REG.
14951  055702  005037  001106                  CLR     @#$TMP4         ;CLEAR ERROR COUNTER
14952  055706  076600                  2$:     MED                     ;MED-READ THE INTERNAL REG.
14953  055710  000000                  10$:    .WORD   0               ;MED-READ CODE
14954  055712  010037  001076                  MOV     R0,@#$TMP0      ;SAVE ITS ORIGINAL CONTENTS
14955  055716  010137  001100                  MOV     R1,@#$TMP1      ;SAVE ADDR. PTR. VALUE
14956  055722  013700  001102                  MOV     @#$TMP2,R0      ;LOAD R0 WITH DATA TO BE WRITTEN
14957  055726  076600                          MED                     ;MED-WRITE THE TEST DATA
14958  055730  000000                  11$:    .WORD   0               ;MED-WRITE CODE
14959  055732  005000                          CLR     R0              ;CLEAR R0
14960  055734  076600                          MED                     ;MED-READ THE DATA BACK
14961  055736  000000                  12$:    .WORD   0               ;MED-READ CODE
14962  055740  010037  001104                  MOV     R0,@#$TMP3      ;SAVE DATA READ FOR COMPARISON
14963  055744  013700  001076                  MOV     @#$TMP0,R0      ;LOAD ORIGINAL DATA IN R0
14964  055750  076600                          MED                     ;MED-WRITE ORG. DATA TO REG.
14965  055752  000000                  13$:    .WORD   0               ;MED-WRITE CODE
14966  055754  023737  001102  001104          CMP     @#$TMP2,@#$TMP3  ;DID DATA READ=DATA WRITTEN?
14967  055762  001412                          BEQ     3$              ;BRANCH IF YES
14968  055764  013737  055736  001100          MOV     @#12$,@#$TMP1   ;SAVE MED-CODE FOR ERROR
14969  055772  022737  000003  001106          CMP     #3,@#$TMP4      ;MAX. ERROR REPORTS YET?
14970  056000  002401                          BLT     14$             ;BRANCH IF YES
14971  056002  104022                          ERROR   22              ;INT. REG. READ BACK WRONG DATA
14972  056004  005237  001106          14$:    INC     @#$TMP4         ;INCREMENT ERROR COUNTER
14973  056010  005137  001102          3$:     COM     @#$TMP2         ;CHANGE DATA PATTERN
14974  056014  013701  001100                  MOV     @#$TMP1,R1      ;RESTORE ADDR. POINTER
```

```
14975  056020  022737  125252  001102          CMP     #125252,@#$TMP2 ;BOTH DATA PATTERNS BEEN USED?
14976  056026  001327                          BNE     2$              ;BRANCH IF NO
14977  056030  005711                          TST     (R1)            ;END OF ADDR. TABLE?
14978  056032  001310                          BNE     1$              ;BRANCH IF NO
14979
14980                                  ;;*****************************************************************
14981                                  ;*TEST 755      MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14982                                  ;*
14983                                  ;*      THIS TEST CHECKS ALL OF THE 'NOP'' OPERATION CODES
14984                                  ;*      TO ENSURE THEY WILL EXECUTE AS NOP'S AND
14985                                  ;*      NOT RESULT IN A PROCESSOR HANG.  THE 'NOPS''
14986                                  ;*      TABLE (TABLE III) HOLDS THESE CODES.
14987                                  ;*      THIS TEST ALSO READS THE PROGRAM COUNTER (R7) VALUES
14988                                  ;*      STORED IN A & B SCRATCH PADS TO SEE THAT THEY
14989                                  ;*      READ PROPERLY.  THE R7 ADDRESSES ARE IN TABLE IV.
14990                                  ;*
14991                                  ;;*****************************************************************
14992  056034                          TST755:
14993  056034  012700  000754                 MOV     #754,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14994  056040  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14995  056042  012701  064404          MEDT3: MOV     #TBL3,R1        ;INITIALIZE NOP TABLE PTR. (R1)
14996  056046  112137  056054          1$:    MOVB    (R1)+,@#10$     ;PLACE FIRST 'NOP-CODE'' AFTER MED
14997                                                                  ;AND POINT R1 TO LAST CODE IN GROUP
14998  056052  076600                  5$:    MED                     ;EXECUTE MED WITH NOP OP-CODE
14999  056054  000000                  10$:   .WORD   0
15000  056056  123711  056054                  CMPB    @#10$,(R1)      ;HAVE ALL NOPS IN THAT GROUP
15001                                                                  ;BEEN TESTED?
15002  056062  103003                          BHIS    6$              ;BRANCH IF YES
15003  056064  005237  056054                  INC     @#10$           ;NEXT NOP IN GROUP
15004  056070  000770                          BR      5$
15005  056072  105721                  6$:    TSTB    (R1)+           ;POINT R1 TO NEXT NOP GROUP
15006  056074  005711                          TST     (R1)            ;HAVE ALL GROUPS BEEN TESTED
15007  056076  001363                          BNE     1$              ;BRANCH IF NO
15008
15009  056100  113737  064425  056112  MEDT4: MOVB    @#R7A+1,@#5$    ;LOAD R7A READ CODE AFTER MED
15010  056106  005000                  4$:    CLR     R0              ;CLEAR R0
15011  056110  076600                          MED                     ;MED READ R7 IN THE ASP
15012  056112  000000                  5$:    .WORD   0               ;READ CODE FOR R7A
15013  056114  020027  056114                  CMP     R0,#5$+2        ;DID R7A READ CORRECTLY?
15014  056120  001411                          BEQ     6$              ;BRANCH IF YES
15015  056122  013737  056112  001100          MOV     @#5$,@#$TMP1    ;SAVE MED-CODE FOR ERROR
15016  056130  012737  056114  001102          MOV     #5$+2,@#$TMP2   ;SAVE DATA EXPECTED
15017  056136  010037  001104                  MOV     R0,@#$TMP3      ;SAVE DATA RECEIVED
15018  056142  104022                          ERROR   22              ;R7A DID NOT READ THE RIGHT VALUE
15019  056144  023727  056112  000047  6$:    CMP     @#5$,#47        ;HAS R7B BEEN CHECKED?
15020  056152  001404                          BEQ     8$              ;BRANCH IF YES
15021  056154  113737  064431  056112          MOVB    @#R7B+1,@#5$    ;LOAD R7B READ CODE AFTER MED
15022  056162  000751                          BR      4$              ;TEST R7 BSP
15023  056164                          8$:
15024
15025
15026                                  ;;*****************************************************************
15027                                  ;*TEST 756      MED TEST - CSP CONSTANTS CHECK
15028                                  ;*
15029                                  ;*      THIS TEST CHECKS THE CONSTANT VALUES LOCATED
15030                                  ;*      IN THE C SCRATCH PAD.  THE CONSTANTS ARE READ
```

```
15031                                   ;*      WITH A MED INSTRUCTION AND COMPARED TO THEIR
15032                                   ;*      EXPECTED VALUE.  THE ADDRESSES OF THESE CONSTANTS
15033                                   ;*      AND THE VALUES EXPECTED ARE IN TABLE VII.
15034                                   ;*
15035                                   ;;**********************************************************************
15036  056164                           TST756:
15037  056164  012700  000755                  MOV     #755,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15038  056170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15039
15040  056172  076600                          MED
15041  056174  000144                          RDFLAG
15042  056176  052700  004000                  BIS     #BIT11,R0       ;SET THE ''CSP INVALID BIT'' IN FLAG REG.
15043  056202  076600                          MED
15044  056204  000344                          WRFLAG
15045  056206  170000                  MEDT10: CFCC                    ;EXECUTE FLT. PT INST. SO FLT. PT.
15046                                                                  ;CONSTANTS ARE LOADED INTO CSP
15047  056210  012701  064536                  MOV     #TBL7,R1        ;SETUP TABLE POINTER
15048  056214  012167  000006          10$:    MOV     (R1)+,1$        ;LOAD MED READ CODE AT 1$
15049  056220  001414                          BEQ     11$             ;BR IF END OF TABLE
15050  056222  005000                          CLR     R0
15051  056224  076600                          MED                     ;READ INTERNAL CONTENTS INTO R0
15052  056226  000000                  1$:     .WORD   0
15053  056230  020021                          CMP     R0,(R1)+        ;WAS THE CONSTANT READ THE ONE EXPECTED
15054  056232  001770                          BEQ     10$             ;BRANCH IF YES
15055  056234  013737  056226  001100          MOV     @#1$,@#$TMP1    ;SAVE MEDCODE FOR ERROR
15056  056242  016137  177776  001102          MOV     -2(R1),@#$TMP2  ;SAVE CONSTANT VALUE EXPECTED
15057  056250  104021                          ERROR   21              ;CSP LOCATION HELD WRONG VALUE
15058  056252                           11$:
15059
15060
15061                                   ;;**********************************************************************
15062                                   ;*TEST 757      MED TEST - MICROBK CHECK OF MICRO-POINTS
15063                                   ;*
15064                                   ;*      THIS TEST USES THE MICROBREAK REGISTER AND THE
15065                                   ;*      INFORMATION IN TABLE V TO CHECK THAT THE
15066                                   ;*      CORRECT MED-FLOW IS ENTERED WHEN EACH
15067                                   ;*      REGISTER IS ACCESSED BY A MED INSTRUCTION.
15068                                   ;*      THE MICROBREAK REG. IS SETUP TO CAUSE A TRAP TO
15069                                   ;*      LOC. 4 WHEN ITS CONTENTS EQUAL THE ADDRESS
15070                                   ;*      OF THE MIRCOWORD BEING EXECUTED.
15071                                   ;*
15072                                   ;*          NOTE:  THE MICRO BREAK - TRAP-TO-4 CAPABILITY
15073                                   ;*                 IS TRIED AT THE BEGINNING OF THE TEST.
15074                                   ;*                 IF IT DOESN'T WORK, AN ERROR IS PRINTED
15075                                   ;*                 AND THE TEST IS SKIPPED
15076                                   ;*
15077                                   ;;**********************************************************************
15078  056252                           TST757:
15079  056252  012700  000756                  MOV     #756,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15080  056256  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15081  056260  012737  000071  177770  MEDT11: MOV     #SWB01,@#UBREAK ;LOAD MICROBK. REG. WITH AN MICRO ADDR.
15082  056266  012737  061104  000004          MOV     #BKROUT,@#4     ;LOAD ADDR. OF MICROBK. ROUTINE IN 4
15083  056274  012737  000340  000006          MOV     #340,@#6        ;LOAD KERNEL PSW - PRIORITY 7 IN 6
15084  056302  005037  061112                  CLR     @#BKFLAG        ;CLEAR MICROBK. TRAP FLAG
15085  056306  076600                          MED                     ;GET WHAMI INTO R0
15086  056310  000022                          RDWHAMI
```

```
15087  056312  052700  001000          BIS     #BIT9,R0        ;SET BIT 9
15088  056316  076600                  MED                     ;MED-WRITE THE WHAMI REG TO
15089  056320  000222          10$:    WRWHAMI                 ;ENABLE MICROBK-TRAP-TO-4
15090  056322  076600                  MED                     ;GET FLAG REGISTER
15091  056324  000144                  RDFLAG
15092  056326  052700  100000          BIS     #BIT15,R0       ;SET BIT 15 IN R0
15093  056332  076600                  MED                     ;MED-WRITE THE FLAG REG TO
15094  056334  000344          11$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15095  056336  000300                  SWAB    R0              ;MICROBK TRAP SHOULD OCCUR ON SWAB
15096  056340  005737  061112          TST     @#BKFLAG        ;DID TRAP TO 4 OCCUR?
15097  056344  001007                  BNE     1$              ;BRANCH IF YES
15098  056346  005037  001076          CLR     @#$TMP0
15099  056352  016737  121513  001100  MOV     SWB01,@#$TMP1   ;SAVE EXPECTED UBREAK ADDR
15100  056360  104015                  ERROR   15              ;MICROBREAK TRAP DIDN'T WORK
15101  056362  000453                  BR      50$             ;SKIP TO END OF TEST
15102
15103  056364  012701  000710    1$:   MOV     #SWB01*10,R1    ;GET CORRECT U-ADDR
15104  056370  076600                  MED                     ;GET LOG CUA REG
15105  056372  000103                  RDLCUA
15106  056374  042700  100007          BIC     #100007,R0      ;GET RID OF IRRELEVANT BITS
15107  056400  020001                  CMP     R0,R1           ;WAS CORRECT UADDR LOGGED?
15108  056402  001401                  BEQ     3$              ;BR IF YES
15109  056404  104025                  ERROR   25              ;CUA CONTAINS INCORRECT U-ADDR
15110  056406  012701  064436    3$:   MOV     #TBL5,R1                ;INTIALIZE TABLE PTR. (R1)
15111  056412  012702  064464          MOV     #TBL6,R2
15112  056416  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP RETURN TO 2$
15113  056422  111137  056460    2$:   MOVB    (R1),@#12$      ;LOAD WRITE CODE AFTER MED
15114  056426  001431                  BEQ     50$             ;BR IF END OF TABLE
15115  056430  011237  177770    4$:   MOV     (R2),@#UBREAK   ;LOAD MICROBK REG. WITH MICROADDR.
15116  056434  005037  061112          CLR     @#BKFLAG        ;CLEAR MICROBK TRAP-TO-4 FLAG
15117  056440  076600                  MED                     ;GET FLAG REGISTER
15118  056442  000144                  RDFLAG
15119  056444  052700  100000          BIS     #BIT15,R0       ;SET BIT 15 IN R0
15120  056450  076600                  MED                     ;MED WRITE TO FLAG REG TO
15121  056452  000344          15$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15122  056454  005000                  CLR     R0              ;IN CASE U-BREAK TRAP DOESN'T OCCOR
15123                                                          ;USUALLY BETTER TO WRITE 0'S
15124  056456  076600                  MED
15125  056460  000000          12$:    .WORD   0
15126  056462  005737  061112          TST     @#BKFLAG        ;DID WE TRAP-TO-4? (FLAG NOT = 0)
15127  056466  001006                  BNE     20$             ;BRANCH IF YES TO NEXT ENTRY
15128  056470  013737  056460  001076  MOV     @#12$,@#$TMP0   ;SAVE MED-CODE FOR ERROR
15129  056476  011237  001100          MOV     (R2),@#$TMP1    ;SAVE EXPECTED U-ADDR FOR ERROR
15130  056502  104015                  ERROR   15              ;MICROBK. TRAP-TO-4 DID NOT OCCUR
15131
15132  056504  105721          20$:    TSTB    (R1)+           ;INCREMENT TO NEXT TABLE
15133  056506  005722                  TST     (R2)+           ;ENTRIES AND
15134  056510  000744                  BR      2$              ;CONTINUE
15135
15136  056512  076600          50$:    MED                     ;GET WHAMI INTO R0
15137  056514  000022                  RDWHAMI
15138  056516  042700  001000          BIC     #BIT9,R0        ; CLEAR BIT 9
15139  056522  076600                  MED                     ;CLEAR THE FLAG REG. TO
15140  056524  000344          13$:    WRFLAG                  ;    DISABLE MICROBK. TRAPPING
15141  056526  076600                  MED                     ;CLEAR THE WHAMI REG. TO
15142  056530  000222          14$:    WRWHAMI                 ;    DISABLE MICROBK. TRAP-TO-4
```

```
15143  056532  012737  056260  001010          MOV     #MEDT11,@#$LPERR        ;RESET LOOP ON ERROR POINTER
15144  056540  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE NORMAL ERROR ROUTINE
15145  056546  012737  000304  177770          MOV     #304,@#UBREAK   ;GENERATE SYNC PULSE ON MED INSTR
15146
15147
15148                                  ;;*********************************************************************
15149                                  ;*TEST 760      PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING
15150                                  ;*      THIS TEST CHECKS THAT THE PROPER PHYSICAL ADDRESS BITS
15151                                  ;*      <17:00> ARE LOGGED UPON ERROR.  THE ERROR IS CAUSED BY
15152                                  ;*      FORCING AN ODD ADDRESS TRAP.  THE ERROR LOG MODE USED
15153                                  ;*      IS 'LOG FIRST'.  ALSO, THE ODD ADDRESS ERROR BITS IN
15154                                  ;*      THE LOG JAM AND CPU ERROR REGISTER ARE CHECKED.
15155                                  ;;*********************************************************************
15156  056554                          TST760:
15157  056554  012700  000757                  MOV     #757,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15158  056560  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15159  056562  012737  056622  000004  1$:     MOV     #2$,@#4         ;SETUP PC FOR ODD ADDR SERVICE
15160  056570  012737  000340  000006          MOV     #340,@#6
15161  056576  012700  100001                  MOV     #BIT15+BIT0,R0  ;SETUP 'LOG FIRST' MODE
15162  056602  076600                          MED
15163  056604  000222                          WRWHAMI
15164  056606  012702  056563                  MOV     #1$+1,R2        ;SAVE ADDRESS OF ODD ADDR. INSTRUCTION
15165  056612  005767  177745                  TST     1$+1            ;DO ODD ADDRESS INSTRUCTION TO FORCE
15166                                                                  ;A JAMUPP & TRAP TO 4
15167  056616  104023                  ERROR   23                      ;*** ODD ADDR. TRAP DID NOT OCCUR
15168  056620  000441                  BR      10$                     ;EXIT TEST
15169  056622  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15170  056624  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW
15171  056632  076600                          MED
15172  056634  000100                          RDLJAM
15173  056636  013701  177766                  MOV     @#CPUERR,R1
15174  056642  032701  000100                  BIT     #BIT6,R1          ;WAS ODD ADDR. ERROR RECORDED BY
15175                                                                  ;THE CPU ERROR REGISTER?
15176  056646  001001                          BNE     3$              ;BRANCH IF YES
15177  056650  104024                          ERROR   24              ;*** CPU ERROR REG. DID NOT
15178                                                                  ;REPORT ODD ADDRESS ERROR
15179                                                                  ;READ THE LOG JAM REGISTER
15180  056652  032700  100004          3$:     BIT     #BIT15+BIT2,R0  ;WAS ODD ADDR. ERROR LOGGED BY LOG JAM
15181  056656  001001                          BNE     4$              ;BRANCH IF YES
15182  056660  104024                          ERROR   24              ;*** LOG JAM REG.  DID NOT LOG
15183                                                                  ;ODD ADDRESS ERROR CORRECTLY
15184
15185  056662  005005                  4$:     CLR     R5              ;CLR ERROR FLAG
15186  056664  076600                          MED                     ;READ THE LOG PBA REGISTER
15187  056666  000102                          RDLPBA
15188  056670  010003                          MOV     R0,R3           ;SAVE RECEIVED PHYS ADDR <15:0>
15189  056672  020002                          CMP     R0,R2           ;WERE BITS <15:00> OF THE PHYSICAL
15190                                                                  ;BUS ADDR. LOGGED CORRECTLY?
15191  056674  001401                          BEQ     5$              ;BRANCH IF YES
15192  056676  005205                          INC     R5              ;SET ERROR FLAG
15193  056700  076600                  5$:     MED                     ;READ THE LOG SERVICE REGISTER
15194  056702  000101                          RDLSERVICE
15195  056704  000300                          SWAB    R0              ;GET 'PBA 17&16' DOWN TO BIT POSITION 0&1
15196  056706  042700  177774                  BIC     #177774,R0
15197  056712  001002                          BNE     11$             ;BR IF PHYS ADDR BITS <17:16> LOGGED CORRECTLY
15198  056714  005705                          TST     R5              ;PREVIOUS ERROR?
```

```
15199  056716  001402                          BEQ     10$             ;BR IF NOT
15200  056720  005001                  11$:    CLR     R1              ;SET UP EXPECTED PA<17:16>
15201  056722  104026                          ERROR   26              ;*** PHYSICAL BUS ADDR. <17:00>
15202                                                                   ;NOT LOGGED CORRECTLY WHEN
15203                                                                   ;ODD ADDRESS TRAP OCCURRED
15204  056724  005000                  10$:    CLR     R0
15205  056726  076600                          MED                     ;DISABLE ''LOG FIRST'' MODE
15206  056730  000222                          WRWHAMI
15207
15208
15209                                  ;;*******************************************************************
15210                                  ;*TEST 761       CHECK DISABLE PARITY ERROR TRAP
15211                                          ;*THIS TEST CHECKS THAT PARITY ERROR TRAPS TO LOCATION 114
15212                                          ;*ARE DISABLED WHEN BIT0 OF THE CACHE CONTROL REGISTER IS
15213                                          ;*SET (=1).  A TRAP TO 114 SHOULD NOT OCCUR AND ERROR
15214                                          ;*INFORMATION SHOULD NOT BE LOGGED IN THE LOG PBA, LOG
15215                                          ;*CACHE DATA, OR LOG TAG DATA REGISTERS.  WRONG PARITY IS
15216                                          ;*WRITTEN INTO A TEST LOCATION TO CAUSE THE PARITY ERROR
15217                                          ;*NEEDED IN THIS TEST.
15218                                  ;;*******************************************************************
15219  056732                          TST761:
15220  056732  012700  000760                  MOV     #760,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15221  056736  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15222
15223  056740  012701  064046                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15224  056744  005711                          TST     (R1)            ;MAKE IT A HIT
15225  056746  012737  000100  177746          MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15226  056754  012711  125252                  MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15227  056760  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE PARITY ERROR TRAPS
15228                                                                   ;AND CLEAR WWP
15229  056766  012737  057026  000114          MOV     #1$,@#114       ;SETUP PARITY ERROR VECTOR
15230  056774  012737  000340  000116          MOV     #340,@#116
15231  057002  005000                          CLR     R0
15232  057004  076600                          MED                     ;CLEAR LOG PBA REGISTER
15233  057006  000302                          WRLPBA
15234  057010  076600                          MED                     ;CLEAR LOG CACHE DATA REGISTER
15235  057012  000306                          WRLDATA
15236  057014  076600                          MED                     ;CLEAR LOG CACHE TAG REGISTER
15237  057016  000307                          WRLTAG
15238  057020  005767  005022                  TST     TLOC1           ;READ TEST LOC0 TO FORCE PARITY ERROR
15239  057024  000406                          BR      2$              ;BRANCH IF NO TRAP OCCURS
15240  057026  012700  000200          1$:     MOV     #200,R0
15241  057032  076600                          MED                     ;CLEAN UP THE CACHE
15242  057034  000352                          352                     ;INITIALIZATION CODE
15243  057036  022626                          CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15244  057040  104030                          ERROR   30              ;*** PARITY TRAP TO 114 OCCURRED
15245                                                                   ;WHEN IT SHOULD HAVE BEEN DISABLED
15246  057042  012700  000200          2$:     MOV     #200,R0
15247  057046  076600                          MED                     ;CLEAN UP THE CACHE
15248  057050  000352                          352                     ;INITIALIZATION CODE
15249  057052  012711  125252                  MOV     #125252,(R1)    ;WRITE BAK GOOD PARITY IN TST LOC
15250  057056  012737  000116  000114          MOV     #116,@#114      ;RESTORE ORIGINAL PARITY HANDLER & PSW
15251  057064  005037  000116                  CLR     @#116
15252  057070  005005                          CLR     R5              ;CLEAR ERROR FLAG
15253  057072  076600                          MED                     ;READ LOG PBA REGISTER
15254  057074  000102                          RDLPBA
```

```
15255  057076  010003                        MOV     R0,R3           ;SAVE COPY
15256                                                                  ;LOG PBA REG. STILL CLEAR?
15257  057100  001401                        BEQ     3$              ;BRANCH IF YES
15258  057102  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15259  057104  076600                3$:     MED                     ;READ LOG CACHE DATA REG.
15260  057106  000106                        RDLDATA
15261  057110  010001                        MOV     R0,R1           ;SAVE COPY
15262                                                                  ;LOG CACHE DATA REG. STILL CLEAR?
15263  057112  001401                        BEQ     4$              ;BRANCH IF YES
15264  057114  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15265  057116  076600                4$:     MED                     ;READ LOG CACHE TAG REG.
15266  057120  000107                        RDLTAG
15267  057122  010002                        MOV     R0,R2           ;SAVE COPY
15268                                                                  ;LOG CACHE TAG REG. STILL CLEAR?
15269  057124  001401                        BEQ     5$              ;BRANCH IF YES
15270  057126  005205                        INC     R5              ;OTHERWISE SET ERROR FLAG
15271  057130  005705                5$:     TST     R5              ;WERE ANY OF LOG REGISTERS CHANGED
15272  057132  001401                        BEQ     6$              ;BRANCH IF NO
15273  057134  104027                        ERROR   27              ;*** ONE OF LOG REGISTERS CHANGED
15274                                                                  ;WHEN ERROR SHOULD NOT HAVE BEEN LOGGED
15275                                                                  ;LOG PBA, LOG DATA & LOG TAG
15276                                                                  ;REGISTER SHOULD BE CLEAR.
15277  057136  005037  177746        6$:     CLR     @#CCR           ;ENABLE PARITY ERROR TRAPS
15278
15279                                ;;*****************************************************************
15280                                ;*TEST 762       CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
15281
15282                                        ;*THIS TEST CHECKS THAT ALL OF THE PARITY ERROR BITS (5,6,7)
15283                                        ;*OF THE MEMORY ERROR REGISTER ARE SET TO "1" WHEN A CACHE
15284                                        ;*PARITY ERROR OCCURS IN THE BACKUP MODE.
15285                                ;;*****************************************************************
15286  057142                        TST762:
15287  057142  012700  000761                MOV     #761,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15288  057146  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15289  057150  012701  064046                MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15290  057154  005711                        TST     (R1)            ;MAKE IT A HIT
15291  057156  012737  000100  177746        MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15292  057164  012711  125252                MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15293  057170  042737  000100  177746        BIC     #WWP,@#CCR      ;CLEAR WWP
15294  057176  012737  057232  000114        MOV     #1$,@#114       ;SETUP NEW TEST HANDLER AT PARITY VECTOR
15295  057204  012737  000340  000116        MOV     #340,@#116
15296  057212  005737  064046                TST     @#TLOC1         ;READ TEST LOC. TO FORCE PARITY ERROR
15297  057216  012700  000200                MOV     #200,R0
15298  057222  076600                        MED                     ;CLEAN UP THE CACHE
15299  057224  000352                        352                     ;INITIALIZATION CODE
15300  057226  104031                        ERROR   31              ;*** PARITY ERROR DID NOT CAUSE TRAP
15301  057230  000405                        BR      2$              ;BRANCH TO 2$
15302  057232  012700  000200        1$:     MOV     #200,R0
15303  057236  076600                        MED                     ;CLEAN UP THE CACHE
15304  057240  000352                        352                     ;INITIALIZATION CODE
15305  057242  022626                        CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15306  057244  022737  000340  177744  2$:     CMP     #000340,@#MEMERR  ;WERE PARITY ERROR BITS (5,6,7) SET
15307                                                                  ;AND CPU ABORT BIT (15) LEFT CLEAR
15308                                                                  ;IN MEMORY ERROR REGISTER?
15309  057252  001403                        BEQ     3$              ;BRANCH IF YES
15310  057254  013700  177744                MOV     @#MEMERR,R0
```

```
15311  057260  104032                          ERROR   32              ;*** MEMORY ERROR REGISTER BITS
15312                                                                  ;WERE SET INCORRECTLY
15313  057262  012737  000116  000114  3$:     MOV     #116,@#114      ;RESTORE OLD PARITY HANDLER PC & PSW
15314  057270  005037  000116                  CLR     @#116
15315
15316                                  ;;*****************************************************************
15317                                  ;*TEST 763       CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
15318
15319                                          ;*THIS TEST CHECKS THAT THE 'UNIBUS TIMEOUT'' BIT (BIT4)
15320                                          ;*GETS SET IN THE CPU ERROR REGISTER WHEN A TIMEOUT OCCURS.
15321                                          ;*A TIMEOUT TRAP IS FORCED BY REFERENCING BUS ADDRESS 760000.
15322                                          ;*THEN AN ODD ADDRESS ERROR IS FORCED AND IT
15323                                          ;*IS CHECKED IF ONLY BIT (6)-ODD ADDRESS ERROR IS SET
15324                                          ;*(IN CPUERR).  THIS CHECKS THAT THE ERROR LOG IS
15325                                          ;*CONTINUOSLY UPDATED IN THE 'LOG CONTINUOUS'' MODE.
15326                                  ;;*****************************************************************
15327  057274                          TST763:
15328  057274  012700  000762                  MOV     #762,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15329  057300  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15330  057302  012737  057324  000004          MOV     #1$,@#4         ;SETUP NEW PC & PSW FOR THE
15331  057310  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT SERVICE ROUTINE
15332  057316  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT TRAP TO 4 BY
15333                                                                  ;REFERENCING NON-EXISTENT ADDRESS
15334  057322  000461                  BR      6$
15335  057324  022626          1$:     CMP     (SP)+,(SP)+             ;RESTORE STACK
15336  057326  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW FOR TIMEOUT
15337  057334  076600                  MED
15338  057336  000100                  RDLJAM
15339  057340  013701  177766                  MOV     @#CPUERR,R1     ;SAVE CPU ERR REG
15340  057344  022701  000020                  CMP     #BIT4,R1        ;DID 'UNIBUS TIMEOUT'' BIT IN CPU ERROR
15341                                                                  ;REGISTER GET SET?
15342  057350  001401                  BEQ     2$                      ;BRANCH IF YES
15343  057352  104033                  ERROR   33                      ;*** 'UNIBUS TIMEOUT'' BIT (BIT4) IN CPU
15344                                                                  ;ERROR REG. DID NOT SET WHEN A
15345                                                                  ;TIMEOUT WAS FORCED
15346                                                                  ;READ THE LOG JAM REGISTER
15347  057354  022700  020200          2$:     CMP     #BIT13+BIT7,R0  ;DID 'UNIBUS TIMEOUT'' BIT (BIT7) SET?
15348  057360  001401                  BEQ     3$                      ;BRANCH IF YES
15349  057362  104033                  ERROR   33                      ;*** 'UNIBUS TIMEOUT'' BIT (BIT7)
15350                                                                  ;DID NOT SET IN LOG JAM REGISTER
15351                                                                  ;WHEN UNIBUS TIMEOUT WAS FORCED
15352  057364  076600          3$:     MED                             ;READ LOG PBA
15353  057366  000102                  RDLPBA
15354  057370  020027  160000                  CMP     R0,#160000      ;WAS PHYS BA LOGGED CORRECTLY?
15355  057374  001403                  BEQ     5$
15356  057376  012701  160000                  MOV     #160000,R1
15357  057402  104020                  ERROR   20                      ;PHYSICAL BUS ADDRESS WAS
15358                                                                  ;LOGGED WRONG ON A UNIBUS
15359                                                                  ;TIMEOUT
15360  057404  012737  057426  000004  5$:     MOV     #4$,@#4         ;SET UP PC,PSW FOR ODD ADDRESS
15361  057412  012737  000340  000006          MOV     #340,@#6
15362  057420  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS ERROR
15363  057424  000420                  BR      6$
15364  057426  022626          4$:     CMP     (SP)+,(SP)+             ;RESTORE STACK
15365  057430  012737  061220  000004          MOV     #BERR,@#4
15366  057436  076600                  MED
```

```
15367  057440  000100                    RDLJAM
15368  057442  013701  177766            MOV     @#CPUERR,R1
15369  057446  022701  000100            CMP     #BIT6,R1        ;ODD ADDR. BUT SET 3
15370  057452  001401                    BEQ     7$
15371  057454  104024                    ERROR   24              ;ODD ADDRESS BIT WAS
15372                                                          ;NOT SET IN THE CPU
15373                                                          ;ERROR REGISTER.  IN LOG
15374                                                          ;CONTINUOUS MADE THE
15375                                                          ;LATEST ERROR SHOULD
15376                                                          ;BE LOGGED
15377  057456  032700  000004        7$:     BIT     #BIT2,R0        ;ODD ADR. BIT SET IN
15378  057462  001001                    BNE     6$              ;LOG JAM?
15379  057464  104024                    ERROR   24              ;ODD ADDRESS BIT WAS
15380                                                          ;NOT SET IN THE LOG
15381                                                          ;JAM REGISTER ON A
15382                                                          ;ODD ADDRESS ERROR
15383  057466  076600                6$:     MED                     ;CHECK IF LAST INTERRUPT VECTOR
15384  057470  000104                    RDLFGINT                ;WAS LOGGED?
15385  057472  120027  000004            CMPB    R0,#4
15386  057476  001401                    BEQ     8$
15387  057500  104036                    ERROR   36              ;LAST ERROR VECTOPR WS NOT LOGGED
15388
15389  057502                        8$:
15390
15391
15392                                ;;**********************************************************************
15393                                ;*TEST 764       CHECK ILLEGAL INTERNAL ADDRESS TRAP
15394
15395                                ;*THIS TEST CHECKS THAT A TRAP OCCURS UPON REFERENCING AN
15396                                ;*ILLEGAL INTERNAL ADDRESS AND THAT ''ILLEGAL INTERNAL ADDRESS''
15397                                ;*BIT (BIT0) OF THE CPU ERROR REGISTER AND BIT5 OF LOG JAM
15398                                ;*REGISTER GET SET.  IT ALSO CHECKS IF THE INTERRUPT VECTOR
15399                                ;*(4) IS SAVED AS THE 'LAST INTERRUPT VECTOR'' IN THE LOG
15400                                ;*FLAG/INTERRUPT REG.
15401                                ;;**********************************************************************
15402  057502                        TST764:
15403  057502  012700  000763            MOV     #763,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15404  057506  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15405  057510  012737  057540  000004    MOV     #1$,@#4         ;SETUP NEW HANDLER PC & PSW
15406  057516  012737  000340  000006    MOV     #340,@#6
15407  057524  005037  177746            CLR     @#CCR
15408  057530  012707  177746            MOV     #CCR,PC         ;ILLEGAL INTERNAL ADDRESS TRAP SHOULD OCCUR
15409  057534  104034                    ERROR   34              ;*** ILLEGAL INTERNAL ADDRESS
15410                                                          ;DID NOT RESULT IN A TRAP
15411  057536  000420                    BR      3$              ;BRANCH TO EXIT IF NO TRAP
15412  057540  022626                1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15413  057542  012737  061220  000004    MOV     #BERR,@#4       ;RESTORE OLD HANDLER PC & PSW
15414  057550  076600                    MED
15415  057552  000100                    RDLJAM
15416  057554  013701  177766            MOV     @#CPUERR,R1
15417  057560  032701  000001            BIT     #BIT0,R1        ;DID ''ILLEGAL INTERNAL ADDRESS'' BIT (0)
15418                                                          ;IN CPU ERROR REGISTER GET SET?
15419  057564  001001                    BNE     2$              ;BRANCH IF YES
15420  057566  104035                    ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS
15421                                                          ;BIT DID NOT SET IN CPU ERROR REG.
15422                                                          ;READ THE LOG JAM REG.
```

```
15423  057570  032700  000040          2$:     BIT     #BIT5,R0        ;DID ''ILLEGAL INTERNAL ADDRESS'' BIT (5)
15424                                                                   ;IN LOG JAM REG. GET SET
15425  057574  001001                          BNE     3$              ;BRANCH IF YES
15426  057576  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS BIT
15427                                                                   ;DID NOT SET IN LOG JAM REG.
15428  057600                          3$:
15429
15430
15431                                  ;;*****************************************************************
15432                                  ;*TEST 765      CHECK LOG SERVICE & MEMERR LOGS LO-HI BYTE & TAG, IN CACHE ABORT MODE
15433                                          ;TEST CHECKS THAT ''LO BYTE PARITY'' ''HI BYTE PARITY'' AND ''TAG PARITY''
15434                                          ;BITS CAN SET IN ''LOG SERVICE'' REGISTERS.  IT IS ALSO
15435                                          ;CHECKED THAT THE PROPER TAG AND DATA BITS GET STORED
15436                                          ;IN THE ''LOG CACHE DATA, ''LOG CACHE TAG/CPU'' AND THE
15437                                          ;''MEMORY ADDRESS REGISTER'' WHEN A PARITY ERROR IS
15438                                          ;FORCED.
15439                                          ;IT IS CHECKED IF THE INSTRUCTION WAS ABORTED AND THE
15440                                          ;LOG FLAG/INTERRUPT REGISTER LOGGED THE LAST INTERRUPT
15441                                          ;VECTOR.
15442                                  ;;*****************************************************************
15443  057600                          TST765:
15444  057600  012700  000764                  MOV     #764,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15445  057604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15446
15447  057606  012737  000201  177746          MOV     #DPTRP+PABORT,@#CCR  ;DISABLE PARITY TRAPS (CACHE)
15448  057614  005037  001062                  CLR     @#$REG0
15449  057620  012701  064046                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOC.
15450  057624  005711                          TST     (R1)            ;MAKE IT A HIT
15451  057626  052737  000100  177746          BIS     #WWP,@#CCR      ;WRITE WRONG PARITY SET
15452  057634  012711  125252                  MOV     #125252,(R1)    ;WRITE TEST LOCATION WITH WRONG PARITY
15453  057640  042737  000100  177746          BIC     #WWP,@#CCR      ;CLEAR WWP
15454  057646  012700  100001                  MOV     #BIT15+BIT0,R0
15455  057652  076600                          MED                     ;ENABLE ''LOG FIRST'' MODE, AND
15456  057654  000222                          WRWHAMI                 ;ERROR LOGGING
15457  057656  042737  000001  177746          BIC     #DPTRP,@#CCR    ;ENABLE CACHE PARITY TRAPS
15458  057664  012737  057712  000114          MOV     #PTRP1,@#114    ;NEW PARITY TRAP SERVICE
15459  057672  016737  004150  001062          MOV     TLOC1,@#$REG0   ;READ TEST LOC, FORCE PARITY ERROR
15460  057700  012700  000200                  MOV     #200,R0
15461  057704  076600                          MED                     ;CLEAN UP THE CACHE
15462  057706  000352                          352                     ;INITIALIZATION CODE
15463  057710  104031                          ERROR   31              ;*** CACHE PARITY ERROR TRAP
15464                                                                   ;DID NOT OCCUR WHEN
15465                                                                   ;TEST LOC WITH BAD PARITY
15466                                                                   ;WAS READ
15467
15468                                                                   ;ENTER HERE IF PARITY TRAP OCCURRED
15469  057712  012700  000200          PTRP1:  MOV     #200,R0
15470  057716  076600                          MED                     ;CLEAN UP THE CACHE
15471  057720  000352                          352                     ;INITIALIZATION CODE
15472  057722  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE CACHE PARITY ERROR TRAPS
15473  057730  012737  000116  000114          MOV     #116,@#114      ;REESTABLISH OLD SERVICE VECTORS
15474  057736  005037  000116                  CLR     @#116
15475  057742  022626                          CMP     (SP)+,(SP)+
15476  057744  005737  001062                  TST     @#$REG0         ;WAS THE INSTRUCTION ABORTED ON
15477                                                                   ;CACHE PARITY ERROR (ABORT MODE)?
15478  057750  001401                          BEQ     1$              ;YES
```

```
15479  057752  104041                 ERROR   41              ;INSTRUCTION HAVING CACHE PARITY
15480                                                          ;ERROR WAS NOT ABORTED, IN THE
15481                                                          ;CACHE ABORT MODE.
15482  057754  076600          1$:    MED                     ;READ THE 'LOG SERVICE' REGISTER
15483  057756  000101                 RDLSERVICE
15484  057760  010004                 MOV     R0,R4           ;COPY
15485  057762  042704  177435         BIC     #^C<LO+HI+TAG+BIT1>,R4  ;MASK ALL BUT LO,HI,TAG BITS
15486  057766  022704  000342         CMP     #342,R4         ;LO,HI ,TAG, CACHE PARITY BITS SET? IN 'SERVICE'
15487  057772  001401                 BEQ     2$              ;YES
15488  057774  104042                 ERROR   42              ;*** 'LO BYTE' PARITY ERROR
15489                                                          ;AND 'TAG' PARITY ERROR BITS
15490                                                          ;WERE NOT LOGGED CORRECTLY IN 'LOG
15491                                                          ;SERVICE' REGISTER, WHEN PARITY
15492                                                          ;ERROR TRAP WAS FORCED.
15493                                                          ;CLEAR BITS ARE ACTIVE.
15494  057776  013700  177744  2$:    MOV     @#MEMERR,R0     ;GET MEM ERR REG
15495  060002  022700  100340         CMP     #HI+LO+TAG+BIT15,R0  ;DID 'LO BYTE' 'HI BYTE' AND 'TAG'
15496                                                          ;PARITY ERROR BITS SET IN
15497                                                          ;THE MEMORY ERROR REGISTER?
15498  060006  001401                 BEQ     3$              ;YES
15499  060010  104043                 ERROR   43              ;*** 'LO BYTE' 'HI BYTE' AND 'TAG' PARITY
15500                                                          ;ERROR BITS DID NOT SET
15501                                                          ;CORRECTLY IN THE MEMORY
15502                                                          ;ERROR REGISTER
15503  060012  076600          3$:    MED                     ;READ 'LOG PBA' REGISTER
15504  060014  000102                 RDLPBA
15505  060016  020027  064046         CMP     R0,#TLOC1       ;DID 'LOG PBA' CONTAIN CORRECT
15506                                                          ;PHYSICAL BUS ADDRESS-WHERE
15507                                                          ;THE PARITY ERROR OCCURRED?
15508  060022  001403                 BEQ     4$              ;YES
15509  060024  012701  064046         MOV     #TLOC1,R1       ;EXPECTED PBA
15510  060030  104020                 ERROR   20              ;*** PHYSICAL BUS ADDRESS
15511                                                          ;(WHERE PARITY ERROR OCCURRED)
15512                                                          ;WAS NOT LOGGED CORRECTLY
15513                                                          ;WHEN CACHE PARITY ERROR WAS FORCED
15514  060032  076600          4$:    MED
15515  060034  000107                 RDLTAG                  ;READ 'LOG CACHE TAG' REGISTER
15516  060036  000300                 SWAB    R0
15517  060040  012701  064046         MOV     #TLOC1,R1       ;SHIFT RIGHT (3 TIMES) THE 16 BIT
15518  060044  000301                 SWAB    R1
15519  060046  106201                 ASRB    R1              ;PHYSICAL BUS ADDRESS OF THE
15520  060050  106201                 ASRB    R1              ;TEST LOCATION
15521  060052  106201                 ASRB    R1
15522  060054  052701  000200         BIS     #BIT7,R1        ;FUDGE TAGE BIT
15523  060060  120100                 CMPB    R1,R0           ;WAS THE CORRECT TAG LOGGED?
15524  060062  001401                 BEQ     5$              ;YES
15525  060064  104017                 ERROR   17              ;TAG BITS WERE NOT LOGGED
15526                                                          ;CORRECTLY, WHEN CACHE
15527                                                          ;PARITY ERROR WAS FORCED
15528  060066  076600          5$:    MED                     ;READ CACHE DATA
15529  060070  000106                 RDLDATA
15530  060072  020027  125252         CMP     R0,#125252      ;CACHE DATA LOGGED CORRECTLY?
15531  060076  001403                 BEQ     6$
15532  060100  012701  125252         MOV     #125252,R1      ;EXPECTED DATA
15533  060104  104016                 ERROR   16
15534
```

```
15535  060106  012700  000001          6$:     MOV     #BIT0,R0        ;SET UP LOG CONTINUOUS
15536  060112  076600                          MED
15537  060114  000222                          WRWHAMI
15538  060116  012737  060130  000004          MOV     #7$,@#4         ;SETUP CPU VECTOR
15539  060124  005737  160000                  TST     @#160000        ;FORCE TIMEOUT & TRAP TO 7$
15540  060130  022626                  7$:     CMP     (SP)+,(SP)+
15541  060132  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE CPU VECTOR
15542  060140  076600                          MED                     ;READ LOG FLAG/INTERRUPT REGISTER
15543  060142  000104                          RDLFGINT
15544  060144  120027  000114                  CMPB    R0,#114         ;DID LO BYTE CONTAIN VECTOR 114?
15545  060150  001403                          BEQ     8$
15546  060152  010037  001062                  MOV     R0,@#$REG0
15547  060156  104036                          ERROR   36              ;LAST INTERRUPT VECTOR WAS NOT
15548                                                                  ;LOGGED CORRECTLY IN FLAG REGISTER
15549                                                                  ;WHEN A CACHE PARITY ERROR WAS
15550                                                                  ;FORCED.
15551  060160                          8$:
15552
15553                                  ;;*****************************************************************
15554                                  ;*TEST 766      CHECK 'LOG FIRST" MODE OF ERROR LOGGING
15555                                          ;*THIS TEST CHECKS THE 'LOG FIRST" MODE OF ERROR LOGGING.
15556                                          ;*THE 'LOG FIRST" MODE IS ENABLED.  THEN A TIME-OUT TRAP
15557                                          ;*IS FORCED, BIT 4 OF CPU ERROR REGISTER SHOULD BE SET.
15558                                          ;*THEN AN ODD ADDRESS TRAP IS FORCED.  HOWEVER, THIS
15559                                          ;*TIME THE ERROR SHOULD NOT BE LOGGED; BIT 6 (ODD
15560                                          ;*ADDRESS) SHOULD NOT BE SET BECAUSE THE ERROR LOG
15561                                          ;*IS LOCKED UP AFTER THE FIRST ERROR.
15562
15563                                          ;*THEN, THE ERROR LOG IS ENABLED (BY SETTING BIT 0 OF
15564                                          ;*WHAMI).  AN ODD ADDRESS ERROR IS FORCED AGAIN AND IT IS
15565                                          ;*CHECKED THAT THIS TIME THE ERROR IS LOGGED, (BIT 6-ODD
15566                                          ;*ADDRESS SHOULD BE SET IN CPU ERROR REGISTER).
15567                                  ;;*****************************************************************
15568  060160                          TST766:
15569  060160  012700  000765                  MOV     #765,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15570  060164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15571
15572  060166  012700  100001                  MOV     #BIT15+BIT0,R0  ;SET UP 'LOG FIRST MODE
15573  060172  076600                          MED
15574  060174  000222                          WRWHAMI
15575  060176  012737  060220  000004          MOV     #1$,@#4         ;SET UP NEW PC & PSW FOR
15576  060204  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT
15577  060212  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT
15578  060216  000462                          BR      5$              ;SKIP TEST IF NO TIMEOUT
15579
15580  060220  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15581                                                                  ;BIT 4 OF CPU ERROR REGISTER
15582                                                                  ;SHOULD HAVE SET
15583  060222  012737  060236  000004          MOV     #2$,@#4         ;SET UP NEW PC FOR ODD ADDRESS
15584  060230  005767  177765                  TST     1$+1            ;FORCE ODD ADDRESS TRAP
15585  060234  000453                          BR      5$              ;SKIP TEST IF NO ODD ADDRESS TRAP
15586
15587  060236  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15588  060240  012737  061220  000004          MOV     #BERR,@#4
15589  060246  076600                          MED
15590  060250  000100                          RDLJAM
```

```
15591  060252  013701  177766                MOV     @#CPUERR,R1
15592  060256  022701  000020                CMP     #BIT4,R1        ;'TIMEOUT'' BIT SHOULD BE STILL
15593                                                                 ;SET, CHECK?
15594  060262  001402                        BEQ     3$
15595  060264  104033                        ERROR   33              ;*** SECOND ERROR (ODD ADDRESS)
15596                                                                 ;UPDATED THE ERROR LOG IN
15597                                                                 ;THE LOG FIRST MODE.  BIT 4
15598                                                                 ;(UNIBUS TIMEOUT) SHOULD BE
15599                                                                 ;STILL SET FROM THE FIRST
15600                                                                 ;ERROR
15601  060266  000436                        BR      5$              ;SKIP THE REST
15602  060270  032700  100004        3$:     BIT     #BIT2+BIT15,R0  ;CHECK THAT ODD ADRES ERROR BITS NOT
15603  060274  001401                        BEQ     6$              ;SET IN LOG JAM.  NOTE LOG FIRST
15604                                                                 ;MODE SHOULD INHIBIT FURTHER
15605                                                                 ;ERROR LOGGING
15606  060276  104037                        ERROR   37              ;ODD ADRESS ERROR BITS GOT SET IN LOG JAM
15607                                                                 ;THEY SHOULD NOT BE SINCE LOG FIRST MODE
15608                                                                 ;INHIBITS ERROR LOGGING AFTER THE FIRST ERROR
15609  060300  012700  100001        6$:     MOV     #BIT15+BIT0,R0  ;ENABLE ERROR LOG AGAIN IN
15610                                                                 ;LOG FIRST MODE
15611  060304  076600                        MED
15612  060306  000222                        WRWHAMI
15613  060310  012737  060332  000004        MOV     #4$,@#4         ;SET UP NEW PC & PSW FOR
15614  060316  012737  000340  000006        MOV     #340,@#6        ;ODD ADDRESS ERROR
15615  060324  005767  177741                TST     3$+1            ;FORCE ODD ADDRESS TRAP
15616  060330  000415                        BR      5$              ;SKIP IF NO TRAP
15617  060332  022626                4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15618                                                                 ;RESTORE OLD PC(4), PSW(6)
15619  060334  012737  061220  000004        MOV     #BERR,@#4
15620  060342  022737  000100  177766        CMP     #BIT6,@#CPUERR  ;THE ERROR LOG FROM PREVIOUS
15621                                                                 ;ERROR SHOULD BE OVER WRITTEN.
15622                                                                 ;ODD ADDRESS BIT SHOULD
15623                                                                 ;BE SET, BECAUSE THE ERROR
15624  060350  001405                        BEQ     5$              ;LOG WAS ENABLED.
15625                                                                 ;OK, IF YES
15626  060352  076600                        MED
15627  060354  000100                        RDLJAM
15628  060356  013701  177766                MOV     @#CPUERR,R1
15629  060362  104040                        ERROR   40              ;THE ERROR LOG WAS NOT UPDATED
15630                                                                 ;(UPON AN ODD ADDRESS ERROR)
15631                                                                 ;AFTER THE LOG WAS ENABLED.
15632                                                                 ;AT THIS FORMAT BIT 6 OF
15633                                                                 ;CPU ERROR REGISTER SHOULD
15634                                                                 ;BE SET.  IT WAS NOT.
15635  060364  012737  061220  000004  5$:     MOV     #BERR,@#4       ;RESTORE OLD PC(4), PSW(6)
15636  060372  012700  000001                MOV     #BIT0,R0
15637  060376  076600                        MED                     ;PUT THE LOGGING BACK INTO
15638  060400  000222                        WRWHAMI                 ;''CONTINUOUS'' MODE
15639
15640                                ;;**********************************************************************
15641                                ;*TEST 767       CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
15642                                ;;**********************************************************************
15643  060402                        TST767:
15644  060402  012700  000766                MOV     #766,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15645  060406  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15646
```

```
15647  060410  012737  060420  000030          MOV     #1$,@#30        ;LOAD EMT VECTOR WITH 1$
15648  060416  104000                          EMT                     ;FIRST INTERRUPT -- EMT
15649  060420  022626                  1$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15650  060422  012737  061620  000030          MOV     #$ERROR,@#30    ;RESTORE VECTOR
15651  060430  012737  060442  000004          MOV     #2$,@#4         ;SET UP CPU VECTOR
15652  060436  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15653  060442  022626                  2$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15654  060444  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15655  060452  076600                          MED                     ;CHECK FLAG
15656  060454  000104                          RDLFGINT
15657  060456  120027  000030                  CMPB    R0,#30          ;EMT VECTOR LAST LOGGED?
15658  060462  001401                          BEQ     3$              ;BR IF YES
15659  060464  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15660                                                                  ;LO BYTE OF LOG FLAG/INT REG S/B=30
15661
15662  060466  012737  060476  000020  3$:     MOV     #4$,@#20        ;LOAD IOT VECTOR WITH 4$
15663  060474  000004                          IOT                     ;SECOND INTERRUPT-SHOULD LOAD LOG FLAG REG
15664  060476  022626                  4$:     CMP     (SP)+,(SP)+     ;CLEANUP STACK
15665  060500  012737  061260  000020          MOV     #$SCOPE,@#20    ;RESTORE IOT VECTOR
15666  060506  012737  060520  000004          MOV     #5$,@#4         ;SET UP CPU VECTOR
15667  060514  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15668  060520  022626                  5$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15669  060522  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15670  060530  076600                          MED                     ;CHECK FLAG
15671  060532  000104                          RDLFGINT
15672  060534  120027  000020                  CMPB    R0,#20          ;IOT VECTOR LAST LOGGED?
15673  060540  001401                          BEQ     6$              ;BR IF YES
15674  060542  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15675                                                                  ;LOW BYTE S/B = 20
15676
15677  060544  012700  000767          6$:     MOV     #$TN-1,R0       ;SET UP FOR MISSED TEST CHECK AND
15678                                                                  ;FULL WORD TEST NUMBER FOR APT
15679
15680                                  .ENABLE AMA
15681
15682
15683                                  .SBTTL  END OF PASS ROUTINE
15684
15685                                  ;;**********************************************************************
15686                                  ;*INCREMENT THE PASS NUMBER ($PASS)
15687                                  ;*IF THERES A MONITOR GO TO IT
15688                                  ;*IF THERE ISN'T JUMP TO INIT
15689
15690  060550                          $EOP:
15691  060550  000004                          SCOPE
15692  060552  005037  001110                  CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
15693  060556  005237  001126                  INC     $PASS           ;;INCREMENT THE PASS NUMBER
15694  060562  042737  100000  001126          BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
15695  060570  005327                          DEC     (PC)+           ;;LOOP?
15696  060572  000001                  $EOPCT: .WORD   1
15697  060574  003027                          BGT     $DOAGN          ;;YES
15698  060576  012737                          MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
15699  060600  000001                  $ENDCT: .WORD   1
15700  060602  060572                          $EOPCT
15701  060604  104401  065103                  TYPE    ,EOP1           ;TYPE 'END PASS #'
15702  060610  013746  001126                  MOV     $PASS,-(SP)     ;SAVE $PASS FOR TYPOUT
```

```
15703  060614  104402                    TYPOC                   ;TYPE PASS NUMBER IN OCTAL
15704  060616  104401  065121            TYPE    ,EOP2           ;TYPE 'ERROR COUNT ='
15705  060622  013746  001012            MOV     $ERTTL,-(SP)    ;SAVE ERROR TOTAL FOR TYPOUT
15706  060626  104402                    TYPOC                   ;TYPE ERROR TOTAL
15707  060630  104401  001115            TYPE    ,$CRLF
15708  060634  013700  000042  $GET42:  MOV     @#42,R0         ;;GET MONITOR ADDRESS
15709  060640  001405                    BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
15710  060642  000005                    RESET                   ;;CLEAR THE WORLD
15711  060644  004710            $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
15712  060646  000240                    NOP                     ;;SAVE ROOM
15713  060650  000240                    NOP                     ;;FOR
15714  060652  000240                    NOP                     ;;ACT11
15715  060654                    $DOAGN:
15716  060654  000137                    JMP     @(PC)+          ;;RETURN
15717  060656  003262            $RTNAD: .WORD   INIT
15718  060660      377     377   000  $ENULL: .BYTE   -1,-1,0      ;;NULL CHARACTER STRING
15719          060664                    .EVEN
```

```
15720                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15721                                           .SBTTL  /  /  /  /  /  /  UTILITIES  /  /  /  /  /  /
15722                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15723
15724                                   .SBTTL  POWER DOWN AND UP ROUTINES
15725
15726                                   ;;*****************************************************************
15727                                   ;POWER DOWN ROUTINE
15728  060664  012737  061036  000024  $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
15729  060672  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15730  060700  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
15731  060702  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
15732  060704  010246                          MOV     R2,-(SP)        ;;PUSH R2 ON STACK
15733  060706  010346                          MOV     R3,-(SP)        ;;PUSH R3 ON STACK
15734  060710  010446                          MOV     R4,-(SP)        ;;PUSH R4 ON STACK
15735  060712  010546                          MOV     R5,-(SP)        ;;PUSH R5 ON STACK
15736  060714  017746  120120                  MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
15737  060720  010637  061042                  MOV     SP,$SAVR6       ;;SAVE SP
15738  060724  012737  060736  000024          MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
15739  060732  000000                          HALT
15740  060734  000776                          BR      .-2             ;;HANG UP
15741
15742                                   ;;*****************************************************************
15743                                   ;POWER UP ROUTINE
15744  060736  012737  061036  000024  $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
15745  060744  013706  061042                  MOV     $SAVR6,SP       ;;GET SP
15746  060750  005037  061042                  CLR     $SAVR6          ;;WAIT LOOP FOR THE TTY
15747  060754  005237  061042          1$:     INC     $SAVR6          ;;WAIT FOR THE INC
15748  060760  001375                          BNE     1$              ;;OF  WORD
15749  060762  011600                          MOV     (SP),R0         ;GET OLD SWR VALUE
15750  060764  076600                          MED                     ;WRITE BACK ORIGINAL SWR VALUE
15751  060766  000226                          WCNSSW                  ;INTO HARDWARE SWITCH REGISTER
15752  060770  012677  120044                  MOV     (SP)+,@SWR      ;;POP STACK INTO @SWR
15753  060774  012605                          MOV     (SP)+,R5        ;;POP STACK INTO R5
15754  060776  012604                          MOV     (SP)+,R4        ;;POP STACK INTO R4
15755  061000  012603                          MOV     (SP)+,R3        ;;POP STACK INTO R3
15756  061002  012602                          MOV     (SP)+,R2        ;;POP STACK INTO R2
15757  061004  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
15758  061006  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
15759  061010  012737  060664  000024          MOV     #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
15760  061016  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15761  061024  104401                          TYPE                    ;REPORT THE POWER FAILURE
15762  061026  061044          $PWRMG: .WORD   $POWER                  ;;POWER FAIL MESSAGE POINTER
15763  061030  012716                  MOV     (PC)+,(SP)              ;;RESTART AT PWRUP
15764  061032  061054          $PWRAD: .WORD   PWRUP                   ;;RESTART ADDRESS
15765  061034  000002                  RTI
15766  061036  000000          $ILLUP: HALT                            ;;THE POWER UP SEQUENCE WAS STARTED
15767  061040  000776                  BR      .-2                     ;; BEFORE THE POWER DOWN WAS COMPLETE
15768  061042  000000          $SAVR6: 0                               ;;PUT THE SP HERE
15769  061044  005015  047520  042527  $POWER: .ASCIZ  <15><12>'POWER'
15770  061052  000122
15771                                           .EVEN
15772
15773  061054  012706  001000          PWRUP:  MOV     #STACK,SP       ;RESET SP
15774  061060  005037  177776                  CLR     @#PSW           ;PRIORITY 0 -- CLEAR CODES
15775  061064  000137  001630                  JMP     @#START         ;RESTART PROGRAM
```

```
15776
15777                                   ; ***************************************************************************
15778                                           .SBTTL 'T'' BIT SERVICE ROUTINE
15779                                   ; ***************************************************************************
15780
15781  061070  062716  000002          TBSER:  ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
15782  061074  042766  000020  000002          BIC     #20,2(SP)       ;TURN OFF THE 'T'' BIT
15783  061102  000006                          RTT                     ;RETURN TO THE CALLING TEST
15784
15785                                   .SBTTL  MICROBREAK TRAP SERVICE ROUTINE
15786                                   ;;*************************************************************
15787                                   ;*      THIS ROUTINE MERELY SETS A FLAG
15788                                   ;*      WHEN THE ROUTINE HAS BEEN ENTERED
15789                                   ;*
15790  061104  005237  061112          BKROUT: INC     BKFLAG          ;SET MICROBREAK FLAG TO
15791                                                                  ;INDICATE TRAP TO 4 OCCURRED
15792  061110  000002                          RTI                     ;RETURN FROM TRAP
15793  061112  000000                  BKFLAG: .WORD   0               ;MICROBREAK TRAP FLAG
15794
15795                                   ; ***************************************************************************
15796                                           .SBTTL  RSVD INSTRUCTION TRAP SERVICE ROUTINE
15797                                   ; ***************************************************************************
15798
15799                                   ;THIS ROUTINE SERVICES UNEXPECTED RESERVED INSTRUCTION TRAP ERRORS
15800                                   ;IT RESULTS IN PRINTING THE ERROR MESSAGE: 'TRAPPED TO 10 PC=XXXXXX''
15801                                   ;WHERE XXXXXX IS THE ADDRESS CONTAINING THE INSTRUCTION WORD THAT
15802                                   ;SPRUNG THE TRAP.  AFTER PRINTING THE ERROR MESSAGE AN ATTEMPT IS
15803                                   ;MADE TO RESTART THE PROGRAM AT THE BEGINNING.
15804
15805                                   ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A
15806                                   ;PREVIOUS RSVD INSTRUCTION TRAP OR AN UNEXPECTED BUS ERROR THE PROGRAM
15807                                   ;WILL HALT.  AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE
15808                                   ;TO THE TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15809
15810                                           ;[SP]   PC+2    OF 2ND TRAP
15811                                           ;[SP]+2 PSW       ''
15812                                           ;[SP]+4 PC+2    OF 1ST TRAP
15813                                           ;[SP]+6 PSW       ''
15814
15815                                           ;LOCATION ''CATERR'' CAN BE EXAMINED TO OBTAIN THE FOLLOWING
15816                                           ;INFORMATION:
15817
15818                                                   ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15819                                                                   ;TRAP (PC AT TIME OF ERROR HALT INDICATES
15820                                                                   ;WHICH OCCURRED FIRST)
15821                                                   ;[CATERR]=2     TWO SUCCESSIVE BUS ERROR TRAPS
15822                                                   ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15823
15824                                   ;THE CONTENTS OF R0 AT THE TIME OF THE
15825                                   ;HALT PROVIDES FURTHER INFORMATION AS TO THE LAST TEST BEING EXECUTED
15826                                   ;WHEN THE TRAPS OCCURRED.
15827
15828                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION
15829                                   ;TESTS TO VERIFY THE RSVD INSTR TRAP MECHANISM PRIOR TO ACTIVATING THE SERVICE
15830                                   ;ROUTINE
15831
```

```
15832  061114  005137  063246          RSVTST: COM     RSVFLG          ;SET RSVD INSTR TRAP TEST FLAG
15833  061120  000002                          RTI                     ;RETURN TO BASIC TEST
15834
15835  061122  005737  063252          RSERR:  TST     @#CATERR        ;ANY PENDING CATASTROPHIC ERRORS
15836  061126  001025                          BNE     INCRSV          ;BE IF YES
15837  061130  105237  063253                  INCB    @#1+CATERR      ;SET RSVD INSTR FLAG
15838  061134  032777  010000  117676          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT ?
15839  061142  001015                          BNE     RESTAR          ;BR IF YES
15840  061144  104401                          TYPE                    ;GO TYPE 'TRAPPED TO 10 PC=''
15841  061146  065232                          RSMSG
15842  061150  011646                  RSBERT: MOV     (SP),-(SP)      ;GET ERROR PC ON STACK FOR PRINTING
15843  061152  104402                          TYPOC                   ;TYPE THE ERROR PC
15844  061154  104401                          TYPE                    ;OUTPUT CR / LF
15845  061156  001115                          $CRLF
15846  061160  005237  001012                  INC     @#$ERTTL        ;COUNT THE ERROR
15847  061164  032777  100000  117646          BIT     #BIT15,@SWR     ;HALT ON ERROR?
15848  061172  001401                          BEQ     RESTAR          ;BR IF NOT
15849  061174  000000                          HALT                    ;HALT ON ERROR--PRESS CONTINUE TO RESTART
15850  061176  000137  003262          RESTAR: JMP     @#INIT          ;GO ATTEMPT RESTART
15851  061202  105237  063253          INCRSV: INCB    @#1+CATERR      ;INCREMENT RSVD INSTR FLAG
15852  061206  000000                          HALT                    ;CATASTROPHIC ERROR HALT
15853  061210  000772                          BR      RESTAR          ;DEPRESSING CONTINUE WILL CAUSE
15854                                                                  ;ATTEMPT TO RESTART.
15855
15856                          ; ***************************************************************************
15857                                  .SBTTL  BUS ERROR TRAP SERVICE ROUTINE
15858                          ; ***************************************************************************
15859
15860                          ;THIS ROUTINE SERVICES UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD ADDRESS
15861                          ;ERRORS, STACK OVERFLOW, AND ILLEGAL INSTRUCTIONS). IT RESULTS IN PRINTING THE
15862                          ;ERROR MESSAGE: ''TRAPPED TO 4 PC =XXXXXX'' WHERE XXXXXX IS THE
15863                          ;CONTENTS OF THE PC WHEN THE TRAP WAS SPRUNG.  AFTER PRINTING THE
15864                          ;ERROR MESSAGE AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
15865                          ;THE BEGINNING.
15866
15867                          ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A PREVIOUS
15868                          ;RSVD INSTR TRAP OR A PREVIOUS BUS ERROR, THE PROGRAM WILL HALT.
15869                          ;AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE TO THE
15870                          ;TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15871
15872                                  ;[SP]   PC+2    OF 2ND TRAP
15873                                  ;[SP]+2 PSW       ''
15874                                  ;[SP]+4 PC+2    OF 1ST TRAP
15875                                  ;[SP]+6 PSW       ''
15876
15877                                  ;LOCATION ''CATERR'' CAN BE EXAMINED TO OBTAIN THE LOLLOWING
15878                                  ;INFORMATION:
15879
15880                                          ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15881                                                          ;TRAP (PC AT TIME OF ERROR HALT
15882                                                          ;INDICATES WHICH OCCURRED FIRST)
15883                                          ;[CATERR]=2     TWO SUCCESSIVE BUS ERRORS
15884                                          ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15885
15886                          ;THE CONTENTS OF R0 AT THE TIME OF
15887                          ;THE HALT PROVIDED FURTHER INFORMATION AS TO THE TEST IN PROGRESS
```

```
15888                                   ;WHEN THE TRAPS OCCURRED.
15889
15890                                   ;THE CONTENTS OF THE SP CAN BE USED TO INDICATE IF STACK OVERFLOW CAUSED
15891                                   ;THE BUSS ERROR TRAP(S) AS SHOWN BELOW:
15892
15893                                           ;400>[SP]>336   YELLOW ZONE
15894                                           ;[SP]=0         RED ZONE
15895
15896                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION TESTS TO
15897                                   ;VERIFY THAT THE BUS ERROR TRAP MECHANISM WORKS PRIOR TO ACTIVATING
15898                                   ;THE SERVICE ROUTINE
15899
15900  061212  005137  063250         BETST:  COM     BERFLG          ;SET BUS ERROR TRAP TEST FLAG
15901  061216  000002                         RTI                     ;RETURN TO BASIC TEST
15902
15903  061220  005737  063252         BERR:   TST     @#CATERR        ;ANY CATASTROPHIC ERRORS PENDING?
15904  061224  001011                         BNE     2$              ;BR IF YES
15905  061226  105237  063252                 INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15906  061232  032777  010000  117600         BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT
15907  061240  001356                         BNE     RESTAR          ;BR IF YES
15908  061242  104401                         TYPE                    ;PRINT 'TRAP TO 4'' MESSAGE
15909  061244  065205                         BEMSG
15910  061246  000740                         BR      RSBERT          ;TYPE REST OF BUS ERROR MESSAGE
15911
15912  061250  105237  063252         2$:     INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15913  061254  000000                         HALT                    ;CATASTROPHIC ERROR HALT-SCHOOLS OUT
15914  061256  000747                         BR      RESTAR          ;DEPRESS CONTINUE TO ATTEMPT RESTART
15915
15916
15917
15918                                   .SBTTL  SCOPE HANDLER ROUTINE
15919
15920                                   ;;**********************************************************************
15921                                   ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
15922                                   ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
15923                                   ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
15924                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
15925                                   ;*SW14=1         LOOP ON TEST
15926                                   ;*SW11=1         INHIBIT ITERATIONS
15927                                   ;*SW09=1         LOOP ON ERROR
15928                                   ;*CALL
15929                                   ;*      SCOPE           ;;SCOPE=IOT
15930
15931  061260                           $SCOPE:
15932  061260  020037  001124                 CMP     R0,@#$TESTN     ;ANY MISSED TESTS ?
15933  061264  001406                         BEQ     10$             ;BR IF NOT
15934  061266  012737  061276  001112         MOV     #12$,@#$ESCAPE  ;NO ERROR LOOPING
15935  061274  104011                         ERROR   11              ;MISSED TESTS ERROR CALL
15936  061276  005037  001112         12$:    CLR     @#$ESCAPE       ;NORMAL ERROR LOOPING
15937  061302  110037  001002         10$:    MOVB    R0,@#$TSTNM     ;INSURE TSTNUM IS CORRECT
15938  061306  032777  002000  117524         BIT     #SW10,@SWR      ;LOOP ON SELECTED TEST?
15939  061314  001411                         BEQ     11$             ;BR IF NO
15940  061316  017737  117516  063242         MOV     @SWR,@#SELTST   ;GET CONTENTS OF SWITCHES
15941  061324  042737  177000  063242         BIC     #177000,@#SELTST        ;MASK OUT SWR<15:9>
15942  061332  020037  063242                 CMP     R0,@#SELTST     ;IS THIS THE SELECTED TEST?
15943  061336  001510                         BEQ     $OVER           ;BR IF YES
```

```
15944  061340                          11$:
15945  061340  032777  040000  117472  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
15946  061346  001104                          BNE     $OVER           ;;YES IF SW14=1
15947                                  ;#####START OF CODE FOR THE XOR TESTER#####
15948  061350  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE 'XOR' TESTER CHANGE
15949                                                                  ;;THIS INSTRUCTION TO A 'NOP' (NOP=240)
15950  061352  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
15951  061356  012737  061376  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
15952  061364  005737  177060                  TST     @#177060        ;;TIME OUT ON XOR?
15953  061370  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15954  061374  000453                          BR      $SVLAD          ;;GO TO THE NEXT TEST
15955  061376  022626                  5$:     CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
15956  061400  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15957  061404  000413                          BR      7$              ;;LOOP ON THE PRESENT TEST
15958  061406                          6$:;#####END OF CODE FOR THE XOR TESTER#####
15959  061406  105737  001003          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
15960  061412  001421                          BEQ     3$              ;;BR IF NO
15961  061414  123737  001015  001003          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
15962  061422  101015                          BHI     3$              ;;BR IF NO
15963  061424  032777  001000  117406          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
15964  061432  001404                          BEQ     4$              ;;BR IF NO
15965  061434  013737  001010  001006  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
15966  061442  000446                          BR      $OVER
15967  061444  105037  001003          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
15968  061450  005037  001110                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
15969  061454  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
15970  061456  032777  004000  117354  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
15971  061464  001011                          BNE     1$              ;;BR IF YES
15972  061466  005737  001126                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
15973  061472  001406                          BEQ     1$              ;;       INHIBIT ITERATIONS
15974  061474  005237  001004                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
15975  061500  023737  001110  001004          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
15976  061506  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
15977  061510  012737  000001  001004  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
15978  061516  013737  061610  001110          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
15979  061524  105237  001002          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
15980  061530  113737  001002  001124          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
15981  061536  011637  001006                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
15982  061542  011637  001010                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
15983  061546  005037  001112                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
15984  061552  112737  000001  001015          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
15985  061560  013777  001002  117254  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
15986  061566  013716  001006                  MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
15987  061572  120037  001002                  CMPB    R0,@#$TSTNM     ;WAS $TSTNM INCREMENTED?
15988  061576  001401                          BEQ     10$             ;BR IF NOT
15989  061600  005200                          INC     R0              ;INCREMENT TEST NUMBER
15990  061602  010037  001124          10$:    MOV     R0,@#$TESTN     ;FIX $TESTN TO BE WORD COUNT, NOT BYTE
15991  061606  000002                          RTI
15992  061610  000200                  $MXCNT: 200                     ;;MAX. NUMBER OF ITERATIONS
15993
15994  061612  005137  063244          SCOPEA: COM     @#SCOFLG        ;THESE TWO INSTRUCTIONS ARE
15995  061616  000002                          RTI                     ;USED IN THE BASIC TESTS TO
15996                                                                  ;VERIFY THE IOT LINKAGE
15997
15998
15999
```

```
 16000                                  .SBTTL  ERROR HANDLER ROUTINE
 16001
 16002                                  ;;**********************************************************************
 16003                                  ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
 16004                                  ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
 16005                                  ;*AND GO TO $ERRTYP ON ERROR
 16006                                  ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
 16007                                  ;*SW15=1         HALT ON ERROR
 16008                                  ;*SW13=1         INHIBIT ERROR TYPEOUTS
 16009                                  ;*SW09=1         LOOP ON ERROR
 16010                                  ;*CALL
 16011                                  ;*       ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER
 16012
 16013  061620                          $ERROR:
 16014  061620  010546                         MOV     R5,-(SP)        ;SAVE R5 ON STACK
 16015  061622  012705  001060                 MOV     #$REGAD,R5      ;GET POINTER
 16016  061626  016625  000004                 MOV     4(SP),(R5)+     ;SAVE ERROR PSW IN $REGAD FOR TYPOUT
 16017  061632  010025                         MOV     R0,(R5)+        ;SAVE R0 FOR TYPEOUTS
 16018  061634  010125                         MOV     R1,(R5)+        ;SAVE R1 IN $REG1
 16019  061636  010225                         MOV     R2,(R5)+        ;SAVE R2 IN $REG2, ETC.
 16020  061640  010325                         MOV     R3,(R5)+
 16021  061642  010425                         MOV     R4,(R5)+
 16022  061644  022715  177777                 CMP     #-1,(R5)        ;IS SP  ALREADY STORED IN $REG5?
 16023  061650  001001                         BNE     10$             ;BR IF YES
 16024  061652  010615                         MOV     SP,(R5)         ;PUT SP IN $REG5 FOR TYPOUT
 16025  061654  012605                 10$:    MOV     (SP)+,R5        ;RESTORE R5
 16026  061656  105237  001003         7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
 16027  061662  001775                         BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
 16028  061664  013777  001002  117150         MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
 16029  061672  005237  001012                 INC     $ERTTL          ;;INC THE ERROR COUNT
 16030  061676  011637  001016                 MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
 16031  061702  162737  000002  001016         SUB     #2,$ERRPC
 16032  061710  117737  117102  001014         MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
 16033  061716  032777  020000  117114         BIT     #BIT13,@SWR     ;;SKIP TYPEOUT IF SET
 16034  061724  001004                         BNE     20$             ;;SKIP TYPEOUTS
 16035  061726  004737  062046                 JSR     PC,$ERRTYP      ;;GO TO USER ERROR ROUTINE
 16036  061732  104401  001115                 TYPE    ,$CRLF
 16037  061736                          20$:
 16038  061736  122737  000001  001140         CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
 16039  061744  001007                         BNE     2$              ;;NO,SKIP APT ERROR REPORT
 16040  061746  113737  001014  061760         MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
 16041  061754  004737  062736                 JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
 16042  061760     000                  21$:    .BYTE   0
 16043  061761     000                          .BYTE   0
 16044  061762  000777                 22$:    BR      22$             ;;APT ERROR LOOP
 16045  061764  005777  117050         2$:     TST     @SWR            ;;HALT ON ERROR
 16046  061770  100001                         BPL     3$              ;;SKIP IF CONTINUE
 16047  061772  000000                         HALT                    ;;HALT ON ERROR!
 16048  061774  032777  001000  117036 3$:     BIT     #BIT09,@SWR     ;;LOOP ON ERROR SWITCH SET?
 16049  062002  001402                         BEQ     4$              ;;BR IF NO
 16050  062004  013716  001010                 MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
 16051  062010  005737  001112         4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
 16052  062014  001402                         BEQ     5$              ;;BR IF NONE
 16053  062016  013716  001112                 MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
 16054  062022                          5$:
 16055  062022  012737  177777  001074         MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
```

```
16056  062030  042766  000020  000002          BIC     #20,2(SP)       ;CLEAR T BIT IN CASE ERROR OCCURED
16057                                                                   ;IN T BIT TESTS
16058  062036  000002                          RTI
16059
16060  062040  005137  063240          ERRA:   COM     @#ERRFLG        ;THESE TWO INSTRUCTIONS ARE USED
16061  062044  000002                          RTI                     ;IN THE BASIC TESTS TO VERIFY THE EMT
16062
16063                                  .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
16064
16065                                  ;;**********************************************************************
16066                                  ;*THIS ROUTINE USES THE ''ITEM CONTROL BYTE'' ($ITEMB) TO DETERMINE WHICH
16067                                  ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE 'ERROR TABLE'' ($ERRTB),
16068                                  ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
16069
16070  062046                          $ERRTYP:
16071  062046  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16072  062052  010046                          MOV     R0,-(SP)        ;;SAVE R0
16073  062054  005000                          CLR     R0              ;;PICKUP THE ITEM INDEX
16074  062056  153700  001014                  BISB    @#$ITEMB,R0
16075  062062  001004                          BNE     1$              ;;IF ITEM NUMBER IS ZERO, JUST
16076                                                                  ;;TYPE THE PC OF THE ERROR
16077  062064  013746  001016                  MOV     $ERRPC,-(SP)    ;;SAVE $ERRPC FOR TYPEOUT
16078                                                                  ;;ERROR ADDRESS
16079  062070  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16080  062072  000426                          BR      6$              ;;GET OUT
16081  062074  005300                  1$:     DEC     R0              ;;ADJUST THE INDEX SO THAT IT WILL
16082  062076  006300                          ASL     R0              ;;      WORK FOR THE ERROR TABLE
16083  062100  006300                          ASL     R0
16084  062102  006300                          ASL     R0
16085  062104  062700  001150                  ADD     #$ERRTB,R0      ;;FORM TABLE POINTER
16086  062110  012037  062120                  MOV     (R0)+,2$        ;;PICKUP ''ERROR MESSAGE'' POINTER
16087  062114  001404                          BEQ     3$              ;;SKIP TYPEOUT IF NO POINTER
16088  062116  104401                          TYPE                    ;;TYPE THE ''ERROR MESSAGE''
16089  062120  000000                  2$:     .WORD   0               ;;''ERROR MESSAGE'' POINTER GOES HERE
16090  062122  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16091  062126  012037  062136          3$:     MOV     (R0)+,4$        ;;PICKUP 'DATA HEADER'' POINTER
16092  062132  001404                          BEQ     5$              ;;SKIP TYPEOUT IF 0
16093  062134  104401                          TYPE                    ;;TYPE THE 'DATA HEADER''
16094  062136  000000                  4$:     .WORD   0               ;;''DATA HEADER'' POINTER GOES HERE
16095  062140  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16096  062144  011000                  5$:     MOV     (R0),R0         ;;PICKUP 'DATA TABLE'' POINTER
16097  062146  001004                          BNE     7$              ;;GO TYPE THE DATA
16098  062150  012600                  6$:     MOV     (SP)+,R0        ;;RESTORE R0
16099  062152  104401  001115                  TYPE    ,$CRLF          ;;''CARRIAGE RETURN'' & 'LINE FEED''
16100  062156  000207                          RTS     PC              ;;RETURN
16101  062160                          7$:
16102  062160  013046                          MOV     @(R0)+,-(SP)    ;;SAVE @(R0)+ FOR TYPEOUT
16103  062162  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16104  062164  005710                          TST     (R0)            ;;IS THERE ANOTHER NUMBER?
16105  062166  001770                          BEQ     6$              ;;BR IF NO
16106  062170  104401  062176                  TYPE    ,8$             ;;TYPE TWO(2) SPACES
16107  062174  000771                          BR      7$              ;;LOOP
16108  062176  020040      000         8$:     .ASCIZ  /  /            ;;TWO(2) SPACES
16109           062202                          .EVEN
16110
16111                                  ; **********************************************************************
```

```
16112                                   .SBTTL PRINT ROUTINES
16113                                   ; ******************************************************************
16114
16115  062202  005137  063236  PRINA:  COM     @#PRIFLG        ;THESE TWO INSTRUCTIONS ARE
16116  062206  000002                  RTI                     ;USED BY THE BASIC TESTS TO VERIFY
16117                                                          ;THE TRAP INSTRUCTION
16118
16119                                   .SBTTL  TYPE ROUTINE
16120
16121                                   ;;******************************************************************
16122                                   ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
16123                                   ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
16124                                   ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
16125                                   ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
16126                                   ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
16127                                   ;*
16128                                   ;*CALL:
16129                                   ;*1) USING A TRAP INSTRUCTION
16130                                   ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
16131                                   ;*OR
16132                                   ;*      TYPE
16133                                   ;*      MESADR_
16134                                   ;*
16135
16136  062210  105737  001057  $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
16137  062214  100002                  BPL     1$              ;;BR IF YES
16138  062216  000000                  HALT                    ;;HALT HERE IF NO TERMINAL
16139  062220  000430                  BR      3$              ;;LEAVE
16140  062222  010046          1$:     MOV     R0,-(SP)        ;;SAVE R0
16141  062224  017600  000002          MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
16142  062230  122737  000001  001140  CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
16143  062236  001011                  BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
16144  062240  132737  000100  001141  BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
16145  062246  001405                  BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
16146  062250  010037  062260          MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
16147  062254  004737  062726          JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
16148  062260  000000          61$:    .WORD   0               ;;MESSAGE ADDRESS
16149  062262  132737  000040  001141  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
16150  062270  001003                  BNE     60$             ;;YES,SKIP TYPE OUT
16151  062272  112046          2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
16152  062274  001005                  BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
16153  062276  005726                  TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
16154  062300  012600          60$:    MOV     (SP)+,R0        ;;RESTORE R0
16155  062302  062716  000002  3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
16156  062306  000002                  RTI                     ;;RETURN
16157  062310  122716  000011  4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
16158  062314  001430                  BEQ     8$
16159  062316  122716  000200          CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
16160  062322  001006                  BNE     5$
16161  062324  005726                  TST     (SP)+           ;;POP  <CR><LF> EQUIV
16162  062326  104401                  TYPE                    ;;TYPE A CR AND LF
16163  062330  001115                  $CRLF
16164  062332  105037  062466          CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
16165  062336  000755                  BR      2$              ;;GET NEXT CHARACTER
16166  062340  004737  062422  5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
16167  062344  123726  001056  6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
```

```
16168  062350  001350                        BNE     2$              ;;IF NO GO GET NEXT CHAR.
16169  062352  013746  001054                MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
16170                                                                 ;;AND THE NULL CHAR.
16171  062356  105366  000001        7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
16172  062362  002770                        BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
16173  062364  004737  062422                JSR     PC,$TYPEC       ;;GO TYPE A NULL
16174  062370  105337  062466                DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
16175  062374  000770                        BR      7$              ;;LOOP
16176
16177                                        ;HORIZONTAL TAB PROCESSOR
16178
16179  062376  112716  000040        8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
16180  062402  004737  062422        9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
16181  062406  132737  000007  062466        BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
16182  062414  001372                        BNE     9$              ;;TAB STOP
16183  062416  005726                        TST     (SP)+           ;;POP SPACE OFF STACK
16184  062420  000724                        BR      2$              ;;GET NEXT CHARACTER
16185  062422  105777  116422        $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
16186  062426  100375                        BPL     $TYPEC
16187  062430  116677  000002  116414        MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
16188  062436  122766  000015  000002        CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
16189  062444  001003                        BNE     1$              ;;BRANCH IF NO
16190  062446  105037  062466                CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
16191  062452  000406                        BR      $TYPEX          ;;EXIT
16192  062454  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
16193  062462  001402                        BEQ     $TYPEX          ;;BRANCH IF YES
16194  062464  105227                        INCB    (PC)+           ;;COUNT THE CHARACTER
16195  062466  000000        $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
16196  062470  000207        $TYPEX: RTS     PC
16197
16198
16199                                        .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
16200
16201                                        ;;**********************************************************************
16202                                        ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
16203                                        ;*OCTAL (ASCII) NUMBER AND TYPE IT.
16204                                        ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
16205                                        ;*CALL:
16206                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16207                                        ;*      TYPOS                   ;;CALL FOR TYPEOUT
16208                                        ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
16209                                        ;*      .BYTE   M               ;;M=1 OR 0
16210                                        ;*                                      ;;1=TYPE LEADING ZEROS
16211                                        ;*                                      ;;0=SUPPRESS LEADING ZEROS
16212                                        ;*
16213                                        ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
16214                                        ;*$TYPOS OR $TYPOC
16215                                        ;*CALL:
16216                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16217                                        ;*      TYPON                   ;;CALL FOR TYPEOUT
16218                                        ;*
16219                                        ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
16220                                        ;*CALL:
16221                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16222                                        ;*      TYPOC                   ;;CALL FOR TYPEOUT
16223
```

```
16224  062472  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
16225  062476  116637  000001  062715          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
16226  062504  112637  062717                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
16227  062510  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
16228  062514  000406                          BR      $TYPON
16229  062516  112737  000001  062715  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
16230  062524  112737  000006  062717          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
16231  062532  112737  000005  062714  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
16232  062540  010346                          MOV     R3,-(SP)        ;;SAVE R3
16233  062542  010446                          MOV     R4,-(SP)        ;;SAVE R4
16234  062544  010546                          MOV     R5,-(SP)        ;;SAVE R5
16235  062546  113704  062717                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
16236  062552  005404                          NEG     R4
16237  062554  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
16238  062560  110437  062716                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
16239  062564  113704  062715                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
16240  062570  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
16241  062574  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
16242  062576  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
16243  062600  000404                          BR      3$              ;;GO DO MSB
16244  062602  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
16245  062604  006105                          ROL     R5
16246  062606  006105                          ROL     R5
16247  062610  010503                          MOV     R5,R3
16248  062612  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
16249  062614  105337  062716                  DECB    $OMODE          ;;TYPE THIS DIGIT?
16250  062620  100016                          BPL     7$              ;;BR IF NO
16251  062622  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
16252  062626  001002                          BNE     4$              ;;TEST FOR 0
16253  062630  005704                          TST     R4              ;;SUPPRESS THIS 0?
16254  062632  001403                          BEQ     5$              ;;BR IF YES
16255  062634  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
16256  062636  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
16257  062642  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
16258  062646  110337  062712                  MOVB    R3,8$           ;;SAVE FOR TYPING
16259  062652  104401  062712                  TYPE    ,8$             ;;GO TYPE THIS DIGIT
16260  062656  105337  062714          7$:     DECB    $OCNT           ;;COUNT BY 1
16261  062662  003347                          BGT     2$              ;;BR IF MORE TO DO
16262  062664  002402                          BLT     6$              ;;BR IF DONE
16263  062666  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
16264  062670  000744                          BR      2$              ;;GO DO THE LAST DIGIT
16265  062672  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
16266  062674  012604                          MOV     (SP)+,R4        ;;RESTORE R4
16267  062676  012603                          MOV     (SP)+,R3        ;;RESTORE R3
16268  062700  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
16269  062706  012616                          MOV     (SP)+,(SP)
16270  062710  000002                          RTI                     ;;RETURN
16271  062712     000                  8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
16272  062713     000                          .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
16273  062714     000          $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
16274  062715     000          $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
16275  062716  000000          $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
16276
16277                          .SBTTL  APT COMMUNICATIONS ROUTINE
16278
16279                          ;;**************************************************************************
```

```
16280  062720  112737  000001  063164  $ATY1:  MOVB    #1,$FFLG        ;;TO REPORT FATAL ERROR
16281  062726  112737  000001  063162  $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
16282  062734  000403                          BR      $ATYC
16283  062736  112737  000001  063164  $ATY4:  MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
16284  062744                          $ATYC:
16285  062744  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
16286  062746  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
16287  062750  105737  063162                  TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
16288  062754  001450                          BEQ     5$              ;;IF NOT:  BR
16289  062756  122737  000001  001140          CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
16290  062764  001031                          BNE     3$              ;;IF NOT:  BR
16291  062766  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
16292  062774  001425                          BEQ     3$              ;;IF NOT:  BR
16293  062776  017600  000004                  MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
16294  063002  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16295  063010  005737  001120          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
16296  063014  001375                          BNE     1$              ;;IF NOT:  WAIT
16297  063016  010037  001134                  MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
16298  063022  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
16299  063024  001376                          BNE     2$
16300  063026  163700  001134                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
16301  063032  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
16302  063034  010037  001136                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
16303  063040  012737  000004  001120          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
16304  063046  000413                          BR      5$
16305  063050  017637  000004  063074  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
16306  063056  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
16307  063064  013746  177776                  MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
16308  063070  004737  062210                  JSR     PC,$TYPE        ;;CALL TYPE MACRO
16309  063074  000000                  4$:     .WORD   0
16310  063076                          5$:
16311  063076  105737  063164          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
16312  063102  001416                          BEQ     12$             ;;IF NOT:  BR
16313  063104  005737  001140                  TST     $ENV            ;;RUNNING UNDER APT?
16314  063110  001413                          BEQ     12$             ;;IF NOT:  BR
16315  063112  005737  001120          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
16316  063116  001375                          BNE     11$             ;;IF NOT:  WAIT
16317  063120  017637  000004  001122          MOV     @4(SP),$FATAL   ;;GET ERROR #
16318  063126  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16319  063134  005237  001120                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
16320  063140  105037  063164          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
16321  063144  105037  063163                  CLRB    $LFLG           ;;CLEAR LOG FLAG
16322  063150  105037  063162                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
16323  063154  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
16324  063156  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
16325  063160  000207                          RTS     PC              ;;RETURN
16326  063162     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
16327  063163     000          $LFLG:  .BYTE   0               ;;LOG FLAG
16328  063164     000          $FFLG:  .BYTE   0               ;;FATAL FLAG
16329          063166                  .EVEN
16330          000200          APTSIZE=200
16331          000001          APTENV=001
16332          000100          APTSPOOL=100
16333          000040          APTCSUP=040
16334
16335                          .SBTTL  TRAP DECODER
```

```
16336
16337                                   ;;**********************************************************************
16338                                   ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE ''TRAP'' INSTRUCTION
16339                                   ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
16340                                   ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
16341                                   ;*GO TO THAT ROUTINE.
16342
16343  063166  010046                   $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
16344  063170  016600  000002                   MOV     2(SP),R0        ;;GET TRAP ADDRESS
16345  063174  005740                           TST     -(R0)           ;;BACKUP BY 2
16346  063176  111000                           MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
16347  063200  006300                           ASL     R0              ;;POSITION FOR INDEXING
16348  063202  016000  063222                   MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
16349  063206  000200                           RTS     R0              ;;GO TO ROUTINE
16350
16351
16352                                   ;;THIS IS USE TO HANDLE THE ''GETPRI'' MACRO
16353
16354  063210  011646                   $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
16355  063212  016666  000004  000002           MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
16356  063220  000002                           RTI                     ;;RESTORE THE PSW
16357
16358                                   .SBTTL  TRAP TABLE
16359
16360                                   ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
16361                                   ;*BY THE ''TRAP'' INSTRUCTION.
16362
16363                                   ;       ROUTINE
16364                                   ;       -------
16365  063222  063210                   $TRPAD: .WORD   $TRAP2
16366  063224  062210                           $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
16367  063226  062516                           $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
16368  063230  062472                           $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
16369  063232  062532                           $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
16370
16371
16372
16373                                   ;FLAGS, CONSTANTS, AND VARIABLES
16374
16375  063234  000000                   BPTLOC: 0                       ;STORES 16 USER DEFINED MAINTENANCE
16376                                                                   ;BREAKPOINTS
16377  063236  000000                   PRIFLG: 0                       ;FLAG USED BY BASIC TESTS FOR TRAP TEST
16378  063240  000000                   ERRFLG: 0                       ;FLAG USED BY BASIC TESTS FOR EMT TEST
16379  063242  000000                   SELTST: 0                       ;STORES SR<8:0> FOR LOOP ON SELECTED TEST
16380  063244  000000                   SCOFLG: 0                       ;USED BY BASIC TESTS FOR IOT TEST
16381  063246  000000                   RSVFLG: 0                       ;FLAG USED BY BASIC TEST OF RSVD INSTR TRAP
16382  063250  000000                   BERFLG: 0                       ;FLAG USED BY BASIC TEST OF BUS ERROR TRAPS
16383  063252  000000                   CATERR: 0                       ;FLAGS USED BY BUS ERROR AND RSVD INSTR TRAP
16384                                                                   ;SERVICE ROUTINES
16385  063254  000000                   ONCE:   0                       ;FLAGS PROGRAM TITLE HAS BEEN PRINTED
16386                                   ;COMMON DATA STRUCTURES AND MISCELLANEOUS TABLES
16387
16388  063256  177400                   OBUF:   177400                  ;DL11 OUTPUT TEST BUFFER
16389  063260  177400                           177400
16390  063262  177400                           177400
16391  063264  177400                           177400
```

```
16392
16393  063266  000004          IBUF:   .BLKW   4               ;DL11 INPUT TEST BUFFER
16394
16395  063276  063322          ATA:    DWTA
16396  063300  064032                  DWTB
16397  063302  064630                  DBTA
16398  063304  064634                  DBTB
16399  063306  063312                  MBUF0
16400  063310  063316                  MBUF1
16401
16402  063312  000000          MBUF0:  0
16403  063314  000000                  0
16404  063316  000000          MBUF1:  0
16405  063320  000000                  0
16406  063322  000000          DWTA:   0
16407  063324  177777                  -1
16408  063326  177400                  177400
16409  063330  000377                  377
16410  063332  125252                  125252
16411  063334  052525          ALUADD: 052525                  ;ALSO SERVES AS NULL ENTRY FOR ALUADD
16412
16413                          ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ADD TEST IN THE
16414                          ;COMBINED INSTRUCTION TESTS
16415
16416  063336  000000                  000000                  ;SRC OP1
16417  063340  000000                  000000                  ;DST OP1
16418  063342  000000                  000000                  ;ANS1
16419  063344  177777                  177777                  ;SRC OP2
16420  063346  177777                  177777                  ;DST OP2
16421  063350  177776                  177776                  ;ANS2
16422  063352  125252                  125252                  ;SRC OP3
16423  063354  052525                  052525                  ;DST OP3
16424  063356  177777                  177777                  ;ANS3
16425  063360  052525                  052525                  ;SRC OP4
16426  063362  125252                  125252                  ;DST OP4
16427  063364  177777                  177777                  ;ANS4
16428  063366  125252                  125252                  ;SRC OP5
16429  063370  125252                  125252                  ;DST OP5
16430  063372  052524                  052524                  ;ANS5
16431  063374  052525                  052525                  ;SRC OP6
16432  063376  052525                  052525                  ;DST OP6
16433  063400  125252                  125252                  ;ANS6
16434  063402  052525                  052525                  ;SRC OP7
16435  063404  125253                  125253                  ;DST OP7
16436  063406  000000                  000000                  ;ANS7
16437  063410  125253                  125253                  ;SRC OP8
16438  063412  052525                  052525                  ;DST OP8
16439  063414  000000          ANDTAB: 000000                  ;ANS8 -- ALSO NULL ENTRY FOR ANDTAB
16440
16441                          ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ''AND'' TESTS IN THE
16442                          ;COMBINED INSTRUCTION EXERCISER TESTS
16443
16444  063416  000000                  000000                  ;SRC OP1
16445  063420  000000                  000000                  ;DEST OP1
16446  063422  000000                  000000                  ;ANS1
16447  063424  177777                  177777                  ;SRC OP2
```

```
16448  063426  177777          177777          ;DST OP2
16449  063430  000000          000000          ;ANS2
16450  063432  000000          000000          ;SRC OP3
16451  063434  177777          177777          ;DST OP3
16452  063436  177777          177777          ;ANS3
16453  063440  177777          177777          ;SRC OP4
16454  063442  000000          000000          ;DST OP4
16455  063444  000000          000000          ;ANS4
16456  063446  125252          125252          ;SRC OP5
16457  063450  125252          125252          ;DST OP5
```

```
16458  063452  000000                    000000                  ;ANS5
16459  063454  052525                    052525                  ;SRC OP6
16460  063456  052525                    052525                  ;DST OP6
16461  063460  000000                    000000                  ;ANS6
16462  063462  125252                    125252                  ;SRC OP7
16463  063464  052525                    052525                  ;DST OP7
16464  063466  052525                    052525                  ;ANS7
16465  063470  052525                    052525                  ;SRC OP8
16466  063472  125252                    125252                  ;DST OP8
16467  063474  125252            ORTAB:  125252                  ;ANS8 -- ALSO NULL ENTRY FOR ORTAB
16468
16469                                    ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU 'OR' TEST IN THE
16470                                    ;COMBINED INSTRUCTION EXERCISER TEST
16471
16472  063476  000000                    000000                  ;SRC OP1
16473  063500  000000                    000000                  ;DEST OP1
16474  063502  000000                    000000                  ;ANS1
16475  063504  177777                    177777                  ;SRC OP2
16476  063506  177777                    177777                  ;DST OP2
16477  063510  177777                    177777                  ;ANS2
16478  063512  000000                    000000                  ;SRC OP3
16479  063514  177777                    177777                  ;DST OP3
16480  063516  177777                    177777                  ;ANS3
16481  063520  177777                    177777                  ;SRC OP4
16482  063522  000000                    000000                  ;DST OP4
16483  063524  177777                    177777                  ;ANS4
16484  063526  125252                    125252                  ;SRC OP5
16485  063530  125252                    125252                  ;DST OP5
16486  063532  125252                    125252                  ;ANS5
16487  063534  052525                    052525                  ;SRC OP6
16488  063536  052525                    052525                  ;DST OP6
16489  063540  052525                    052525                  ;ANS6
16490  063542  125252                    125252                  ;SRC OP7
16491  063544  052525                    052525                  ;DST OP7
16492  063546  177777                    177777                  ;ANS7
16493  063550  052525                    052525                  ;SRC OP8
16494  063552  125252                    125252                  ;DST OP8
16495  063554  177777           ALUSUB:  177777                  ;ANS8 -- ALSO NULL ENTRY FOR ALUSUB
16496
16497                                    ;THIS TABLE OF 8 ENTRIES IS USED BYTHE ALU SUB TEST IN THE
16498                                    ;COMBINED INSTRUCTION EXERCISER TESTS
16499
16500  063556  000000                    000000                  ;SRC OP1
16501  063560  000000                    000000                  ;DST OP1
16502  063562  000000                    000000                  ;ANS1
16503  063564  177777                    177777                  ;SRC OP2
16504  063566  177777                    177777                  ;DST OP2
16505  063570  000000                    000000                  ;ANS2
16506  063572  125252                    125252                  ;SRC OP3
16507  063574  052525                    052525                  ;DST OP3
16508  063576  125253                    125253                  ;ANS3
16509  063600  052525                    052525                  ;SRC OP4
16510  063602  125252                    125252                  ;DST OP4
16511  063604  052525                    052525                  ;ANS4
16512  063606  125252                    125252                  ;SRC OP5
16513  063610  125252                    125252                  ;DST OP5
```

```
16514  063612  000000                    000000                  ;ANS5
16515  063614  052525                    052525                  ;SRC OP6
16516  063616  052525                    052525                  ;DST OP6
16517  063620  000000                    000000                  ;ANS6
16518  063622  052525                    052525                  ;SRC OP7
16519  063624  125253                    125253                  ;DST OP7
16520  063626  052526                    052526                  ;ANS7
16521  063630  125253                    125253                  ;SRC OP8
16522  063632  052525                    052525                  ;DST OP8
16523  063634  125252                    125252                  ;ANS8
16524
16525  063636  005702            INSTAB: TST     R2      ;BEGINNING OF INSTRUCTION TABLE OF INSTRUCTIONS
16526  063640  005002                    CLR     R2      ;THAT  TEST BUT SERVICE IN VARIOUS ROM LOCATIONS
16527  063642  005102                    COM     R2
16528  063644  005202                    INC     R2
16529  063646  005302                    DEC     R2
16530  063650  005502                    ADC     R2
16531  063652  005602                    SBC     R2
16532  063654  006202                    ASR     R2
16533  063656  006302                    ASL     R2
16534  063660  105002                    CLRB    R2
16535  063662  105102                    COMB    R2
16536  063664  105202                    INCB    R2
16537  063666  105302                    DECB    R2
16538  063670  105502                    ADCB    R2
16539  063672  105502                    ADCB    R2
16540  063674  105602                    SBCB    R2
16541  063676  105702                    TSTB    R2
16542  063700  106202                    ASRB    R2
16543  063702  106302                    ASLB    R2
16544  063704  151302                    BISB    (R3),R2
16545  063706  074302                    XOR     R3,R2
16546  063710  121302                    CMPB    (R3),R2
16547  063712  131302                    BITB    (R3),R2
16548  063714  141302                    BICB    (R3),R2
16549  063716  111302                    MOVB    (R3),R2
16550  063720  021302                    CMP     (R3),R2
16551  063722  031302                    BIT     (R3),R2
16552  063724  041302                    BIC     (R3),R2
16553  063726  051302                    BIS     (R3),R2
16554  063730  006702                    SXT     R2
16555  063732  005402                    NEG     R2
16556  063734  161302                    SUB     (R3),R2
16557  063736  020312                    CMP     R3,(R2)
16558  063740  030312                    BIT     R3,(R2)
16559  063742  120312                    CMPB    R3,(R2)
16560  063744  131302                    BITB    (R3),R2
16561  063746  005712                    TST     (R2)
16562  063750  105712                    TSTB    (R2)
16563  063752  021312                    CMP     (R3),(R2)
16564  063754  031312                    BIT     (R3),(R2)
16565  063756  121312                    CMPB    (R3),(R2)
16566  063760  131312                    BITB    (R3),(R2)
16567  063762  061302                    ADD     (R3),R2
16568  063764  000302                    SWAB    R2
16569  063766  160302                    SUB     R3,R2
```

```
16570  063770  060302                  ADD     R3,R2
16571  063772  010302                  MOV     R3,R2
16572  063774  011302                  MOV     (R3),R2
16573  063776  110302                  MOVB    R3,R2
16574  064000  006102                  ROL     R2
16575  064002  106102                  ROLB    R2
16576  064004  105402                  NEGB    R2
16577  064006  102400                  BVS     .+2
16578  064010  102000                  BVC     .+2
16579  064012  000005                  RESET
16580  064014  020302                  CMP     R3,R2
16581  064016  030302                  BIT     R3,R2
16582  064020  040302                  BIC     R3,R2
16583  064022  120302                  CMPB    R3,R2
16584  064024  130302                  BITB    R3,R2
16585  064026  140302                  BICB    R3,R2
16586  064030  150302                  BISB    R3,R2
16587
16588  064032  000000          DWTB:   0                       ;ALSO SERVES AS INSTAB TABLE TERMINATOR
16589  064034  000001                  1
16590  064036  000400                  400
16591  064040  177401                  177401
16592  064042  052526                  52526
16593  064044  125253                  125253
16594                           ;*      MED TEST TABLES
16595
16596  064046  000000          TLOC1:  .WORD   0
16597  064050  000000          PSWHOL: .WORD   0
16598  064052  000000          TABBEG: .WORD   0
16599  064054  000000          TABEND: .WORD   0
16600  064056  000040          STGBLK: .BLKW   40
16601  064156  000000          VADR:   .WORD   0
16602  064160  000000          PA1716: .WORD   0
16603  064162  000000          PA1500: .WORD   0
16604  064164  000000          TLOC2:  .WORD   0
16605
16606                           ;*
16607                           ;*      TABLE II
16608                           ;*
16609                           ;*      FOLLOWING IS A TABLE OF INTERNAL REGISTER OPERATION CODES
16610                           ;*      USED FOR TESTING THE MED INSTRUCTION.  LABELS CORRESPOND
16611                           ;*      TO REGISTER NAMES, THE HIGH BYTE IS THE READ OPERATION
16612                           ;*      CODE, THE LOW BYTE THE WRITE CODE.
16613                           ;*              NOTE:  WHEN ADDING OR DELETING
16614                           ;*              ENTRIES IN THIS TABLE, CHECK DUAL
16615                           ;*              ADDRESING TEST TO SEE THAT THE 'SCRATCH
16616                           ;*              PAD LIMITS'' ARE MAINTAINED.
16617                           ;*
16618
16619  064166                  TBL2:
16620
16621  064166                  ASP1:                           ;A SCRATCH PAD - LO
16622  064166  201     001    R1A:    .BYTE   201,001          ;LOBYTE, HIBYTE=WRITE CODE, READ CODE
16623  064170  202     002    R2A:    .BYTE   202,002
16624  064172  203     003    R3A:    .BYTE   203,003
16625  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   GNWHAM:  .BYTE   222,022
064230   223   023   CNSTSW:  .BYTE   223,023
064232   226   026   CNSSW:   .BYTE   226,026
064234   227   027   CNSCDR:  .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
064326   272   072   FAC0.1:  .BYTE   272,072
064330   273   073   FAC0.2:  .BYTE   273,073
064332   274   074   FAC0.4:  .BYTE   274,074
064334   275   075   FAC0.5:  .BYTE   275,075
064336   276   076   FEA:     .BYTE   276,076
064340   277   077   BSP2:    FDST0:  .BYTE   277,077

064342                CSP1:
064342   300   100   LJAM:    .BYTE   300,100          ;C SCRATCH PAD
064344   301   101   LSERV:   .BYTE   301,101
```

```
16682  064346      302      102      LPBA:   .BYTE   302,102
16683  064350      303      103      LCUA:   .BYTE   303,103
16684  064352      304      104      LFGIN:  .BYTE   304,104
16685  064354      305      105      LWHAM:  .BYTE   305,105
16686  064356      307      107      LTAG:   .BYTE   307,107
16687  064360      310      110      CNSC0:  .BYTE   310,110
16688  064362      311      111      CNSC1:  .BYTE   311,111
16689  064364      312      112      CNSC2:  .BYTE   312,112
16690  064366      313      113      CST200: .BYTE   313,113
16691  064370      316      116      CSP2:   CNST0:  .BYTE   316,116
16692  064372      224      024      RT1A:   .BYTE   224,024
16693  064374      225      025      RT2A:   .BYTE   225,025
16694  064376      264      064      RT1B:   .BYTE   264,064
16695  064400      265      065      RT2B:   .BYTE   265,065
16696  064402   000000                       .WORD   0
16697
16698                                 ;*
16699                                 ;*      TABLE III
16700                                 ;*
16701                                 ;*      THE FOLLOWING IS A LIST OF 'NOP'' OPERATION CODES
16702                                 ;*      THAT WILL BE USED WITH A MED IN MED TEST 3 TO
16703                                 ;*      ENSURE THAT A MED WITH THESE CODES WILL NOT HANG.
16704
16705  064404                         TBL3:
16706  064404      120      137      NOPS:   .BYTE   120,137         ;GROUP A
16707  064406      145      145              .BYTE   145,145         ;GROUP B
16708  064410      150      151              .BYTE   150,151         ;GROUP C
16709  064412      156      177              .BYTE   156,177         ;GROUP D
16710  064414      320      343              .BYTE   320,343         ;GROUP E
16711  064416      353      357              .BYTE   353,357         ;FROUP G
16712  064420   000000                       .WORD   0               ;A 0 TERMINATES TABLE
16713
16714                                 ;*
16715                                 ;*      TABLE IV
16716                                 ;*
16717                                 ;*      THE LIST BELOW CONTAINS THOSE OPERATION CODES
16718                                 ;*      CORRESPONDING TO THE INTERNAL REGISTERS WHICH MUST
16719                                 ;*      BE TESTED SEPERATELY BECAUSE THEY ARE READ-ONLY,
16720                                 ;*      WRITE-ONLY, OR USED IN MACRO CODE EXECUTION, ETC. . .
16721
16722  064422                         TBL4:
16723  064422      200      000      R0A:    .BYTE   200,000         ;LOBYTE,HYBYTE - WRITE CODE, READ CODE
16724  064424      207      007      R7A:    .BYTE   207,007         ;0 REPLACES ANY NON EXSISTENT CODES
16725  064426      240      040      R0B:    .BYTE   240,040         ;EXCEPT IN THE CASE OF R0A
16726  064430      247      047      R7B:    .BYTE   247,047
16727  064432      314      114      CNST2:  .BYTE   314,114
16728  064434      317      117      CNST1:  .BYTE   317,117
16729                                 ;*      TABLE V
16730                                 ;*
16731  064436                         TBL5:
16732
16733  064436      306               LCDTA:  .BYTE   306             ;THIS TABLE CONTAINS THE OPERATION
16734  064437      106                       .BYTE   106             ;CODES OF THOSE INTERNAL REGISTERS
16735  064440      315               MD:     .BYTE   315             ;WHICH MUST BE TESTED USING THE
16736  064441      115                       .BYTE   115             ;MICROBREAK REGISTER.  THEIR
16737  064442      267               CNSCTL: .BYTE   267             ;ASSOCIATED MICRO-ADDRESSES ARE IN
```

```
16738  064443  067
16739  064444  140
16740  064445  141
16741  064446  142
16742  064447  143
16743  064450  344
16744  064451  144
16745  064452  345
16746  064453  146
16747  064454  346
16748  064455  147
16749  064456  347
16750  064457  351
16751  064460  152
16752  064461  352
16753  064462  153
16754  064463  000
16755
16756
16757
16758
16759  064464
16760
16761  064464  003330
16762  064466  003150
16763  064470  003375
16764  064472  003271
16765  064474  003240
16766  064476  003224
16767  064500  003160
16768  064502  003161
16769  064504  003170
16770  064506  003171
16771  064510  003344
16772  064512  003320
16773  064514  003345
16774  064516  003340
16775  064520  003350
16776  064522  003341
16777  064524  003351
16778  064526  003355
16779  064530  003720
16780  064532  003724
16781  064534  003721
16782
16783
16784
16785
16786
16787  064536
16788
16789  064536  000100  077600
16790  064542  000101  000010
16791  064546  000102  020000
16792  064552  000103  000004
16793  064556  000104  050000
```

```
                .BYTE   067             ;THE NEXT TABLE
JAM:            .BYTE   140
SERV:           .BYTE   141
PBA:            .BYTE   142
CUA:            .BYTE   143
FLAG:           .BYTE   344
                .BYTE   144
DREG:           .BYTE   345
REV:            .BYTE   146
SREG:           .BYTE   346
COUNT:          .BYTE   147
NUA:            .BYTE   347
RES:            .BYTE   351
DCS0:           .BYTE   152
                .BYTE   352             ;INIT REG
DCS1:           .BYTE   153
                .BYTE   0               ;TABLE TERMINATOR
                .EVEN

;*      TABLE VI
;*
TBL6:

ULCDTA:         .WORD   3330            ;THIS TABLE CONTAINS THE MICRO-ADDRESSES
                .WORD   3150            ;WHICH ARE LOADED INTO THE MICROBREAK
UMD:            .WORD   3375            ;REG. TO TEST THE OPERATION CODES
                .WORD   3271            ;CONTAINED IN THE PRECEEDING TABLE.
UCNSCTL:        .WORD   3240
                .WORD   3224
UJAM:           .WORD   3160
USERV:          .WORD   3161
UPBA:           .WORD   3170
UCUA:           .WORD   3171
UFLAG:          .WORD   3344
                .WORD   3320
UDREG:          .WORD   3345
UREV:           .WORD   3340
USREG:          .WORD   3350
UCOUNT:         .WORD   3341
UNUA:           .WORD   3351
URES:           .WORD   3355
UDCS0:          .WORD   3720
UINIT:          .WORD   3724
UDCS1:          .WORD   3721

;*      TABLE VII
;*
;*      THIS TABLE HOLDS THE OPERATION CODES AND THE CONSTANT
;*      VALUE EXPECTED FOR CERTAIN INTERNAL REGISTERS.
TBL7:

CLJAM:          .WORD   100,77600
CLSERV:         .WORD   101,10
CLPBA:          .WORD   102,20000
CLCUA:          .WORD   103,4
CLFGIN:         .WORD   104,50000
```

```
16794  064562  000105  054000          CLWHAM: .WORD   105,54000
16795  064566  000107  024000          CLTAG:  .WORD   107,24000
16796  064572  000110  177400          CCNSC0: .WORD   110,177400
16797  064576  000111  177600          CCNSC1: .WORD   111,177600
16798  064602  000112  100000          CCNSC2: .WORD   112,100000
16799  064606  000113  000200          CCST200:.WORD   113,200
16800  064612  000114  000002          CCNST2: .WORD   114,2
16801  064616  000116  000000          CCNST0: .WORD   116,0
16802  064622  000117  000001          CCNST1: .WORD   117,1
16803  064626  000000                          .WORD   0
16804
16805                                  .EVEN
16806  064630                          DBTA:
16807  064630      000     377     252 .BYTE   000,377,252,125
16808  064633      125
16809  064634                          DBTB:
16810  064634      000     001     120 .BYTE.  000,001,120,253
16811  064637      253
16812
16813                                  ;MESSAGE TABLES
16814
16815  064640                          EM1:
16816  064640                          EM2:
16817  064640  027523  020102  051504  EM4:    .ASCII  'S/B DST '
16818  064646  020124
16819  064650  040527  020123  051504  EM7:    .ASCII  'WAS DST '
16820  064656  020124
16821  064660  042040  051505  004524  EM6:    .ASCII  ' DEST'<HT>
16822  064666  024040  051111  004451  EM5:    .ASCIZ  ' (IR)'<HT>' TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16823  064674  052040  051505  004524
16824  064702  024040  041520  004451
16825  064710  024040  050123  004451
16826  064716  050050  053523  000051
16827  064724  027523  020102  042522  EM10:   .ASCIZ  'S/B RES WAS RES DST OP  SRC OP  TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16828  064732  020123  040527  020123
16829  064740  042522  020123  051504
16830  064746  020124  050117  020040
16831  064754  051123  020103  050117
16832  064762  020040  042524  052123
16833  064770  020011  050050  024503
16834  064776  020011  051450  024520
16835  065004  024011  051520  024527
16836  065012      000
16837  065013      123 041057  051440  EM3:    .ASCIZ  'S/B SP'<HT>'WAS SP'<HT>' (IR)'<HT>' TEST'<HT>' (PC)'<HT>'(PSW)'
16838  065020  004520  040527  020123
16839  065026  050123  020011  044450
16840  065034  024522  020011  042524
16841  065042  052123  020011  050050
16842  065050  024503  024011  051520
16843  065056  024527      000
16844  065061      011 020011  051511  DH2:    .ASCIZ  <HT><HT>' IS R3'
16845  065066  051040  000063
16846  065072  004411  044440  020123  DH4:    .ASCIZ  <HT><HT>' IS R5'
16847  065100  032522      000
16848  065103      015 042412  042116  ECP1:   .ASCIZ  <15><12>'END PASS # '
16849  065110  050040  051501  020123
```

```
16850  065116  020043     000
16851  065121     011  051105  047522  EOP2:   .ASCIZ <HT>'ERROR COUNT = '
16852  065126  020122  047503  047125
16853  065134  020124  020075     000
16854  065141     015  041412  045521  IDENT1: .ASCIZ <15><12>'CQKDAC  KD11-K BASIC LOGIC TESTS'<15><12>
16855  065146  040504  004503  042113
16856  065154  030461  045455  041040
16857  065162  051501  041511  046040
16858  065170  043517  041511  052040
16859  065176  051505  051524  005015
16860  065204     000
16861  065205     015  052012  040522  BEMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 4 PC = '
16862  065212  050120  042105  052040
16863  065220  020117  020064  041520
16864  065226  036440  000040
16865  065232  005015  051124  050101  RSMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 10 PC = '
16866  065240  042520  020104  047524
16867  065246  030440  020060  041520
16868  065254  036440  000040
16869  065260  042524  052123  020123  EM11:   .ASCIZ  'TESTS SKIPPED'
16870  065266  045523  050111  042520
16871  065274  000104
16872  065276  020040  041520  042411  DH11:   .ASCIZ  '' PC''<HT>''EXPCTD''<HT>''ACTUAL''<HT>''(TEST #'S)''
16873  065304  050130  052103  004504
16874  065312  041501  052524  046101
16875  065320  024011  042524  052123
16876  065326  021440  051447  000051
16877  065334  042515  020104  044504  EM12:   .ASCIZ  /MED DID NOT ABORT IN USER MODE/
16878  065342  020104  047516  020124
16879  065350  041101  051117  020124
16880  065356  047111  052440  042523
16881  065364  020122  047515  042504
16882  065372     000
16883  065373     115  042105  042440  EM13:   .ASCIZ  /MED EXECUTED IN USER MODE/
16884  065400  042530  052503  042524
16885  065406  020104  047111  052440
16886  065414  042523  020122  047515
16887  065422  042504     000
16888  065425     115  042105  041440  EM14:   .ASCIZ  /MED CHANGED PSW/
16889  065432  040510  043516  042103
16890  065440  050040  053523     000
16891  065445     115  041511  047522  EM15:   .ASCIZ  /MICROBREAK TRAP-TO-4 DID NOT OCCUR/
16892  065452  051102  040505  020113
16893  065460  051124  050101  052055
16894  065466  026517  020064  044504
16895  065474  020104  047516  020124
16896  065502  041517  052503  000122
16897  065510  047514  041507  040525  EM17:   .ASCIZ  /LOGCUA LOGGED WRONG/
16898  065516  046040  043517  042507
16899  065524  020104  051127  047117
16900  065532  000107
16901  065534  051503  020120  047503  EM21:   .ASCIZ  /CSP CONSTANT WRONG/
16902  065542  051516  040524  052116
16903  065550  053440  047522  043516
16904  065556     000
16905  065557     102  042101  042040  EM22:   .ASCIZ  /BAD DATA READ BY A MED/
```

```
16906  065564  052101  020101  042522
16907  065572  042101  041040  020131
16908  065600  020101  042515  000104
16909  065606  047516  047440  042104  EM23:   .ASCIZ  /NO ODD PC TRAP/
16910  065614  050040  020103  051124
16911  065622  050101     000
16912  065625     117  042104  040440  EM24:   .ASCIZ  /ODD ADR. BIT NOT SET IN CPU ERR REG OR LOG JAM/
16913  065632  051104  020056  044502
16914  065640  020124  047516  020124
16915  065646  042523  020124  047111
16916  065654  041440  052520  042440
16917  065662  051122  051040  043505
16918  065670  047440  020122  047514
16919  065676  020107  040512  000115
16920  065704  044120  051531  041040  EM26:   .ASCIZ  /PHYS BA LOGGED WRONG/
16921  065712  020101  047514  043507
16922  065720  042105  053440  047522
16923  065726  043516     000
16924  065731     103  041501  042510  EM27:   .ASCIZ  /CACHE PARITY ERROR LOGGED IN BAKUP MODE/
16925  065736  050040  051101  052111
16926  065744  020131  051105  047522
16927  065752  020122  047514  043507
16928  065760  042105  044440  020116
16929  065766  040502  052513  020120
16930  065774  047515  042504     000
16931  066001     103  041501  042510  EM30:   .ASCIZ  /CACHE PARITY TRAPPED WHEN DISABLED/
16932  066006  050040  051101  052111
16933  066014  020131  051124  050101
16934  066022  042520  020104  044127
16935  066030  047105  042040  051511
16936  066036  041101  042514  000104
16937  066044  047111  052123  027122  EM41:   .ASCIZ  /INSTR. NOT ABORTED IN CACHE ABORT MODE/
16938  066052  047040  052117  040440
16939  066060  047502  052122  042105
16940  066066  044440  020116  040503
16941  066074  044103  020105  041101
16942  066102  051117  020124  047515
16943  066110  042504     000
16944  066113     115  046505  051117  EM32:   .ASCIZ  /MEMORY ERR REG INCORRECT/
16945  066120  020131  051105  020122
16946  066126  042522  020107  047111
16947  066134  047503  051122  041505
16948  066142  000124
16949  066144  044524  042515  052517  EM33:   .ASCIZ  /TIMEOUT BIT NOT SET IN CPU ERR REG OR LOG JAM/
16950  066152  020124  044502  020124
16951  066160  047516  020124  042523
16952  066166  020124  047111  041440
16953  066174  052520  042440  051122
16954  066202  051040  043505  047440
16955  066210  020122  047514  020107
16956  066216  040512  000115
16957  066222  047516  044440  046114  EM34:   .ASCIZ  /NO ILLEGAL INTERNAL ADR TRAP/
16958  066230  043505  046101  044440
16959  066236  052116  051105  040516
16960  066244  020114  042101  020122
16961  066252  051124  050101     000
```

```
16962  066257     111  052116  047122  EM35:   .ASCIZ  /INTRNAL ADR ERR BIT NOT SET IN CPU ERR REG OR LOG JAM/
16963  066264  046101  040440  051104
16964  066272  042440  051122  041040
16965  066300  052111  047040  052117
16966  066306  051440  052105  044440
16967  066314  020116  050103  020125
16968  066322  051105  020122  042522
16969  066330  020107  051117  046040
16970  066336  043517  045040  046501
16971  066344     000
16972  066345     114  051501  020124  EM36:   .ASCIZ  'LAST INTR/TRAP VECTOR NOT LOGGED IN FLAG REG'
16973  066352  047111  051124  052057
16974  066360  040522  020120  042526
16975  066366  052103  051117  047040
16976  066374  052117  046040  043517
16977  066402  042507  020104  047111
16978  066410  043040  040514  020107
16979  066416  042522  000107
16980  066422  047514  020107  044506  EM37:   .ASCIZ  /LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR/
16981  066430  051522  020124  047515
16982  066436  042504  042040  042111
16983  066444  047040  052117  044440
16984  066452  044116  041111  052111
16985  066460  042440  051122  051117
16986  066466  046040  043517  040440
16987  066474  052106  051105  043040
16988  066502  051111  052123  042440
16989  066510  051122  051117     000
16990  066515     105  051122  051117  EM40:   .ASCIZ  /ERROR LOG WAS NOT REENABLED, ODD ADR BIT CLR IN CPUERR/
16991  066522  046040  043517  053440
16992  066530  051501  047040  052117
16993  066536  051040  042505  040516
16994  066544  046102  042105  020054
16995  066552  042117  020104  042101
16996  066560  020122  044502  020124
16997  066566  046103  020122  047111
16998  066574  041440  052520  051105
16999  066602  000122
17000  066604  047516  041440  041501  EM31:   .ASCIZ  /NO CACHE PARITY TRAP/
17001  066612  042510  050040  051101
17002  066620  052111  020131  051124
17003  066626  050101     000
17004  066631     114  020117  020046  EM42:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE/
17005  066636  044510  041040  052131
17006  066644  020105  020046  040524
17007  066652  020107  040520  044522
17008  066660  054524  041040  052111
17009  066666  020123  047516  020124
17010  066674  042523  020124  047111
17011  066702  046040  043517  051440
17012  066710  051105  044526  042503
17013  066716     000
17014  066717     114  020117  020046  EM43:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN MEM ERR REG/
17015  066724  044510  041040  052131
17016  066732  020105  020046  040524
17017  066740  020107  040520  044522
```

```
17018  066746  054524  041040  052111
17019  066754  020123  047516  020124
17020  066762  042523  020124  047111
17021  066770  046440  046505  042440
17022  066776  051122  051040  043505
17023  067004     000
17024  067005     103  041501  042510  EM45:    .ASCIZ  /CACHE TAG LOGGED WRONG/
17025  067012  052040  043501  046040
17026  067020  043517  042507  020104
17027  067026  051127  047117  000107
17028  067034  040503  044103  020105  EM16:    .ASCIZ  /CACHE DATA LOGGED WRONG/
17029  067042  040504  040524  046040
17030  067050  043517  042507  020104
17031  067056  051127  047117  000107
17032  067064  044505  020123  042523  EMEIS1: .ASCIZ  'EIS SET COND CODES WRONG'
17033  067072  020124  047503  042116
17034  067100  041440  042117  051505
17035  067106  053440  047522  043516
17036  067114     000
17037  067115     105  051511  043440  EMEIS2: .ASCIZ  'EIS GAVE WRONG RESULT'
17038  067122  053101  020105  051127
17039  067130  047117  020107  042522
17040  067136  052523  052114     000
17041  067143     101  052125  026517  EM46:    .ASCIZ  'AUTO-INCREMENT (DECREMT) DID NOT OCCUR IN EIS'
17042  067150  047111  051103  046505
17043  067156  047105  020124  042050
17044  067164  041505  042522  052115
17045  067172  020051  044504  020104
17046  067200  047516  020124  041517
17047  067206  052503  020122  047111
17048  067214  042440  051511     000
17049  067221     040  051520  004527  DHEIS1: .ASCII  ' PSW'<HT>'REG-WAS-REG+1'<HT>'REG-S/B-REG+1'<HT>
17050  067226  042522  026507  040527
17051  067234  026523  042522  025507
17052  067242  004461  042522  026507
17053  067250  027523  026502  042522
17054  067256  025507  004461
17055  067262  020040  041520  020011  DH46:    .ASCIZ  '  PC'<HT>' (IR)'<HT>' TEST'
17056  067270  044450  024522  020011
17057  067276  042524  052123     000
17058  067303     040  050040  004503  DH15:    .ASCIZ  /  PC/<HT>/MEDCODE MICROBK REG./
17059  067310  042515  041504  042117
17060  067316  020105  044515  051103
17061  067324  041117  020113  042522
17062  067332  027107     000
17063  067335     040  050040  004503  DH17:    .ASCIZ  /  PC/<HT>/MEDCODE EXPECTD RECEIVD/
17064  067342  042515  041504  042117
17065  067350  020105  054105  042520
17066  067356  052103  020104  042522
17067  067364  042503  053111  000104
17068  067372  020040  041520     000  DH23:    .ASCIZ  /  PC/
17069  067377     040  050040  004503  DH24:    .ASCIZ  /  PC/<HT>/CPUERR/<HT>/LOGJAM/
17070  067404  050103  042525  051122
17071  067412  046011  043517  040512
17072  067420  000115
17073  067422  020040  041520  043011  DH25:    .ASCIZ  /  PC/<HT>/FLGREG/
```

```
17074  067430  043514  042522  000107
17075  067436  020040  041520  036011  DH26:   .ASCIZ  '  PC'<HT>'<17:16>-S/B PA-<15:0>  <17:16>-WAS PA-<15:0>'
17076  067444  033461  030472  037066
17077  067452  051455  041057  050040
17078  067460  026501  030474  035065
17079  067466  037060  020040  030474
17080  067474  035067  033061  026476
17081  067502  040527  020123  040520
17082  067510  036055  032461  030072
17083  067516  000076
17084  067520  020040  041520  046011  DH27:   .ASCIZ  /  PC/<HT>/LOGPBA/<HT>/LOGDATA/<HT>/LOGTAG/
17085  067526  043517  041120  004501
17086  067534  047514  042107  052101
17087  067542  004501  047514  052107
17088  067550  043501     000
17089  067553     040  050040  004503  DH32:   .ASCIZ  /  PC/<HT>/MEMERR/
17090  067560  042515  042515  051122
17091  067566     000
17092  067567     040  050040  004503  DH42:   .ASCIZ  /  PC/<HT>/LOGSERVCE/
17093  067574  047514  051507  051105
17094  067602  041526  000105
17095  067606  020040  041520  042411  DH44:   .ASCIZ  /  PC/<HT>/EXPCT/<HT>/RECVD/
17096  067614  050130  052103  051011
17097  067622  041505  042126     000
17098          067630          .EVEN
17099  067630  001016  001076  001100  DT15:   .WORD   $ERRPC,$TMP0,$TMP1,0
17100  067636  000000
17101  067640  001016  001100  001102  DT21:   .WORD   $ERRPC,$TMP1,$TMP2,$REG0,0
17102  067646  001062  000000
17103  067652  001016  001100  001102  DT22:   .WORD   $ERRPC,$TMP1,$TMP2,$TMP3,0
17104  067660  001104  000000
17105  067664  001016  000000          DT23:   .WORD   $ERRPC,0
17106  067670  001016  001064  001062  DT24:   .WORD   $ERRPC,$REG1,$REG0,0
17107  067676  000000
17108  067700  001016  001062  000000  DT25:   .WORD   $ERRPC,$REG0,0
17109  067706  001016  001064  001066  DT26:   .WORD   $ERRPC,$REG1,$REG2,$REG0,$REG3,0
17110  067714  001062  001070  000000
17111  067722  001016  001070  001064  DT27:   .WORD   $ERRPC,$REG3,$REG1,$REG2,0
17112  067730  001066  000000
17113  067734  001060  001066  001070  DTEIS1: .WORD   $REGAD,$REG2,$REG3,$REG1,$REG4
17114  067742  001064  001072
17115  067746  001016  001076  001062  DT46:   .WORD   $ERRPC,$TMP0,$REG0,0
17116  067754  000000
17117
17118  067756     000     000          DF15:   .BYTE   0,0
17119  067760     000     000     000  DF17:   .BYTE   0,0,0
17120          067764          .EVEN
17121  067764                  DT1:
17122  067764                  DT2:
17123  067764                  DT4:
17124  067764  001072          DT10:   .WORD   $REG4
17125  067766  001070          DT7:    .WORD   $REG3
17126  067770  001066          DT6:    .WORD   $REG2
17127  067772  001064  001062  001016  DT5:    .WORD   $REG1,$REG0,$ERRPC,$REG5,$REGAD,0
17128  070000  001074  001060  000000
17129  070006  001072  001070  001064  DT3:    .WORD   $REG4,$REG3,$REG1,$REG0,$ERRPC,$REGAD,0
```

```
17130  070014  001062  001016  001060
17131  070022  000000
17132  070024  001016  001124  001062  DT11:   .WORD   $ERRPC,$TESTN,$REG0,0
17133  070032  000000
17134          000001                  .END
```

```
ABASE  = 000000            933
ACDW1  = 000000            933
ACDW2  = 000000            933
ACPUOP= 000000            933     948
ADDW0  = 000000            933
ADDW1  = 000000            933
ADDW10= 000000            933
ADDW11= 000000            933
ADDW12= 000000            933
ADDW13= 000000            933
ADDW14= 000000            933
ADDW15= 000000            933
ADDW2  = 000000            933
ADDW3  = 000000            933
ADDW4  = 000000            933
ADDW5  = 000000            933
ADDW6  = 000000            933
ADDW7  = 000000            933
ADDW8  = 000000            933
ADDW9  = 000000            933
ADEVCT= 000000            933     939
ADEVM  = 000000            933
AENV   = 000000            933     944
AENVM  = 000000            933     945
AFATAL= 000000            933     936
ALUADD  063334          14007   14012   16411#
ALUSUB  063554          14067   14072   16495#
AMADR1= 000000            933
AMADR2= 000000            933
AMADR3= 000000            933
AMADR4= 000000            933
AMAMS1= 000000            933
AMAMS2= 000000            933
AMAMS3= 000000            933
AMAMS4= 000000            933
AMSGAD= 000000            933     941
AMSGLG= 000000            933     942
AMSGTY= 000000            933     935
AMTYP1= 000000            933
AMTYP2= 000000            933
AMTYP3= 000000            933
AMTYP4= 000000            933
ANDTAB  063414          14127   14132   16439#
APASS  = 000000            933     938
APRIOR= 000000            933
APTCSU= 000040          16149   16333#
APTENV= 000001          16038   16142   16289   16331#
APTSIZ= 000200          16330#
APTSPO= 000100          16144   16291   16332#
ASP1    064166          16621#
ASP2    064254          16649#
ASWREG= 000000            933     946
ATA     063276           2641    2691    2703    2719    2751    2764    2812    2833    2852    2864    2874    2886    2897
                         7632    7643    7656    7667    7680    7692    7705    7716    7729    7741    7754    7773    7794
                         7815    7836    7865    7894    7995    7909    7923    7924    7938    7952    7981    8010    8011
                         8025    8039    8040    8054    8068    8090    8112    8134    8772    8784    8824    8836    8871
```

```
                    9096    9110    9133    9147    9248    9262    9285    9299    9386    9418    9490    9532    9767
                    9772    9821    9826    9870    9935   11084   11127   11171   14409   14429   14481   14501   14548
                   16395#
ATESTN= 000000      933     937
AUNIT = 000000      933     940
AUSWR = 000000      933     947
AVECT1= 000000      933
AVECT2= 000000      933
A002    001636     1285#   1289
A015    002106     1451    1456#
A017    002200     1494    1501#
A020    002246     1523    1528#
A021    002276     1545    1550#
A023    002364     1586    1591#
A032    002570     1691    1696#
A033    002630     1713    1718#
A034    002670     1734    1739#
A041    003034     1815    1820#
A046    003234     1913    1918#
BASIC   003246     1919    1927#
BEMSG   065205    15909   16861#
BERFLG  063250     4468*   4474*  15900*  16382#
BERR    061220     4506   12315   12319   13183   13221   13260   13277   13329   13389   13427   13458   13490   14917
                  15144   15170   15336   15365   15413   15541   15588   15619   15635   15654   15669   15903#
BETST   061212     4465   15900#
BIT0  = 000001      815#    1222    2297   15161   15417   15454   15535   15572   15609   15636
BIT00 = 000001      805#     815
BIT01 = 000002      804#     814
BIT02 = 000004      803#     813
BIT03 = 000010      802#     812
BIT04 = 000020      801#     811
BIT05 = 000040      800#     810
BIT06 = 000100      799#     809
BIT07 = 000200      798#     808
BIT08 = 000400      797#     807
BIT09 = 001000      796#     806   15963   16048
BIT1  = 000002      814#    2958   15485
BIT10 = 002000      795#   10026
BIT11 = 004000      794#   10896   15042   15970
BIT12 = 010000      793#   11708
BIT13 = 020000      792#   12762   15347   16033
BIT14 = 040000      791#   13902   15945
BIT15 = 100000      790#   14853   15092   15119   15161   15180   15454   15495   15572   15602   15609   15847
BIT2  = 000004      813#    3690   15180   15377   15602
BIT3  = 000010      812#    4199
BIT4  = 000020      811#    4924   15340   15592
BIT5  = 000040      810#    1226    5797   15423
BIT6  = 000100      809#    1221    1224    6649   15174   15369   15620
BIT7  = 000200      808#    1223    1225    7505   15347   15522
BIT8  = 000400      807#    8320
BIT9  = 001000      806#    9241   15087   15138
BKFLAG  061112    15084*   15096   15116*  15126   15790*  15793#
BKROUT  061104    15082    15790#
BPTLOC  063234     2297    2958    3690    4199    4924    5797    6649    7505    8320    9241   10026   10896   11708
                  12762   13902   14853   16375#
BPTVEC= 000014      822#
```

```
BSP1    064256          16651#
BSP2    064340          16677#
BT001   001630           1275#
BT002   001634           1275    1283#
BT003   001646           1285    1297#   1304
BT004   001662           1300    1310#   1319
BT005   001700           1315    1325#   1334
BT006   001716           1330    1340#   1347
BT007   001730           1344    1353#   1362
BT010   001744           1358    1368#   1377
BT011   001762           1374    1383#   1393
BT012   002002           1390    1399#   1408
BT013   002022           1405    1414#   1425
BT014   002046           1422    1431#   1440
BT015   002070           1437    1446#   1454    1461
BT016   002116           1458    1467#   1479
BT017   002144           1475    1485#   1499    1508
BT020   002220           1503    1514#   1526    1532
BT021   002256           1529    1538#   1548    1555
BT022   002310           1552    1561#   1571
BT023   002334           1568    1577#   1589    1595
BT024   002374           1592    1601#   1610
BT025   002420           1607    1616#   1624
BT026   002440           1621    1629#   1637
BT027   002460           1634    1643#   1651
BT030   002500           1648    1657#   1665
BT031   002520           1662    1670#   1678
BT032   002540           1675    1684#   1694    1700
BT033   002600           1697    1706#   1716    1722
BT034   002640           1719    1727#   1737    1743
BT035   002700           1740    1748#   1757
BT036   002720           1754    1762#   1771
BT037   002740           1768    1776#   1785
BT040   002760           1782    1791#   1802
BT041   003006           1799    1807#   1818    1824
BT042   003044           1821    1830#   1840
BT043   003064           1837    1846#   1857
BT044   003106           1854    1863#
BT045   003134           1870    1873    1878#   1888
BT046   003162           1885    1894#   1916    1922
CATERR  063252          15835   15837*  15851*  15903   15905*  15912*  16383#
CCNSC0  064572          16796#
CCNSC1  064576          16797#
CCNSC2  064602          16798#
CCNST0  064616          16801#
CCNST1  064622          16802#
CCNST2  064612          16800#
CCR     =  177746        1220#  15225*  15227*  15277*  15291*  15293*  15407*  15408   15447*  15451*  15453*  15457*  15472*
CCST20  064606          16799#
CITST   013412           4609    4621#
CLCUA   064552          16792#
CLFGIN  064556          16793#
CLJAM   064536          16789#
CLPBA   064546          16791#
CLSERV  064542          16790#
CLTAG   064566          16795#
```

```
CLWHAM  064562          16794#
CNSCDR  064234          16641#
CNSCTL  064442          16737#
CNSC0   064360          16687#
CNSC1   064362          16688#
CNSC2   064364          16689#
CNSSW   064232          16640#
CNSTSW  064230          16639#
CNST0   064370          16691#
CNST1   064434          16728#
CNST2   064432          16727#
COUNT   064455          16748#
CPUERR= 177766           1219#  15173  15339  15368  15416  15591  15620  15628
CR    = 000015            730#  16188  16198  16861  16865
CRLF  = 000200            731#  16159  16198
CSP1    064342          16679#
CSP2    064370          16691#
CST200  064366          16690#
CUA     064447          16742#
DBTA    064630           2770   3316   3328   3366   3378   3566   3574   9629   9649   9668   9679   9692   9703
                         9717   9738   9765   9792   9819   9846   9868  11019  11039  11060  11082  11104  11126
                        11148  11170  11256  16397  16806#
DBTB    064634           3341   3353   3391   3403   3442   3494   3655   3695  16398  16809#
DCS0    064460          16751#
DCS1    064462          16753#
DDISP = 177570            737#    902
DF15    067756           1041  17118#
DF17    067760           1068   1074  17119#
DHEIS1  067221           1197   1204  17049#
DH11    065276           1015  16872#
DH15    067303           1039  17058#
DH17    067335           1066   1072  17063#
DH2     065061            980  16844#
DH23    067372           1021   1027   1033   1078   1113   1120   1141   1176  17068#
DH24    067377           1085   1134   1148   1162   1169  17069#
DH25    067422           1155  17073#
DH26    067436           1099  17075#
DH27    067520           1106  17084#
DH32    067553           1127   1190  17089#
DH4     065072            990  16846#
DH42    067567           1183  17092#
DH44    067606           1046   1053   1060   1092  17095#
DH46    067262           1211  17055#
DISPLA  001042            902#  15985*  16028*
DISPRE  000174            837#
DPTRP = 000001           1222#  15227  15447  15457  15472
DREG    064452          16745#
DSWR  = 177570            736#    901
DTEIS1  067734           1198   1205  17113#
DT1     067764            976  17121#
DT10    067764           1011  17124#
DT11    070024           1016  17132#
DT15    067630           1040  17099#
DT2     067764            981  17122#
DT21    067640           1067  17101#
DT22    067652           1073  17103#
```

```
DT23    067664          1022    1028    1034    1079    1114    1121    1142    1177   17105#
DT24    067670          1047    1054    1061    1086    1093    1135    1149    1163    1170   17106#
DT25    067700          1128    1156    1184    1191   17108#
DT26    067706          1100   17109#
DT27    067722          1107   17111#
DT3     070006           986   17129#
DT4     067764           991   17123#
DT46    067746          1212   17115#
DT5     067772           996   17127#
DT6     067770          1001   17126#
DT7     067766          1006   17125#
DWTA    063322          2322    2324    2525    2571    2575    2594    2598    2643    2697    2715    2746    2806    2832
                        2851    2880    2913    3166    3167    3178    3184    3203    3204    3205    3215    3221    3240
                        3241    3242    3252    3258    3277    3278    3289    3295    3518    3530    3540    3552    6682
                        7631    7655    7679    7704    7728    7772    7793    7814    7835    7864    7893    7922    7951
                        7980    8009    8038    8067    8089    8111    8133    8630    8893    8925    8956    8988    9021
                        9058    9095    9132    9169    9206    9247    9284    9321    9353    9385    9417   10505   10532
                       10587   10614   10641   10667   10694   10749   10778   16395   16406#
DWTB    064032          2547    2549    2903    3416    3468    3602    3610    3637    3673    8603   16396   16588#
EA017   002166          1491    1492    1493    1496#
EA020   002242          1525#
EA021   002272          1547#
EA023   002360          1583    1584    1585    1588#
EA032   002564          1693#
EA033   002624          1715#
EA034   002664          1736#
EA041   003030          1817#
EA046   003230          1915#
EMEIS1  067064          1196   17032#
EMEIS2  067115          1203   17037#
EMTVEC= 000030           825#
EM1     064640           974   16815#
EM10    064724          1009   16827#
EM11    065260          1014   16869#
EM12    065334          1020   16877#
EM13    065373          1026   16883#
EM14    065425          1032   16888#
EM15    065445          1038   16891#
EM16    067034          1045   17028#
EM17    065510          1091   16897#
EM2     064640           979   16816#
EM21    065534          1065   16901#
EM22    065557          1071   16905#
EM23    065606          1077   16909#
EM24    065625          1084   16912#
EM26    065704          1059    1098   16920#
EM27    065731          1105   16924#
EM3     065013           984   16837#
EM30    066001          1112   16931#
EM31    066604          1119   17000#
EM32    066113          1126   16944#
EM33    066144          1133   16949#
EM34    066222          1140   16957#
EM35    066257          1147   16962#
EM36    066345          1154   16972#
EM37    066422          1161   16980#
```

```
EM4     064640            989   16817#
EM40    066515           1168   16990#
EM41    066044           1175   16937#
EM42    066631           1182   17004#
EM43    066717           1189   17014#
EM45    067005           1052   17024#
EM46    067143           1210   17041#
EM5     064666            994   16822#
EM6     064660            999   16821#
EM7     064650           1004   16819#
EOP1    065103          15701   16848#
EOP2    065121          15704   16851#
ERRA    062040           4423   16060#
ERRFLG  063240           4425*   4430*  16060*  16378#
ERRVEC= 000004            818#  14904*  14917*  15950   15951*  15953*  15956*
EX002   001640           1287#
E001    001632           1277#
E003    001656           1297    1298    1299    1302#
E004    001674           1312    1313    1314    1317#
E005    001712           1327    1328    1329    1332#
E006    001724           1346#
E007    001740           1360#
E010    001756           1373    1376#
E011    001776           1389    1392#
E012    002016           1407#
E013    002042           1424#
E014    002064           1439#
E016    002136           1472    1473    1474    1477#
E022    002330           1565    1566    1567    1570#
E024    002414           1609#
E025    002434           1620    1623#
E026    002454           1633    1636#
E027    002474           1650#
E030    002514           1664#
E031    002534           1677#
E035    002714           1756#
E036    002734           1770#
E037    002754           1784#
E040    003002           1801#
E042    003060           1839#
E043    003102           1856#
E044    003130           1872#
E045    003156           1887#
E1015A  002102           1453#
E2002   001644           1291#
E2015   002112           1460#
E2017   002206           1505#
E2020   002252           1531#
E2021   002304           1554#
E2023   002370           1594#
E2032   002574           1699#
E2033   002634           1721#
E2034   002674           1742#
E2041   003040           1823#
E2046   003242           1921#
FAC0.0  064324          16671#
```

```
FAC0.1  064326          16672#
FAC0.2  064330          16673#
FAC0.4  064332          16674#
FAC0.5  064334          16675#
FAC1.0  064236          16642#
FAC1.1  064240          16643#
FAC1.2  064242          16644#
FAC1.3  064244          16645#
FAC1.4  064246          16646#
FAC1.5  064250          16647#
FAC2.0  064272          16658#
FAC2.1  064274          16659#
FAC2.2  064276          16660#
FAC2.3  064300          16661#
FAC2.4  064302          16662#
FAC2.5  064304          16663#
FAC3.0  064202          16628#
FAC3.1  064204          16629#
FAC3.2  064206          16630#
FAC3.3  064210          16631#
FAC3.4  064212          16632#
FAC3.5  064214          16633#
FDST0   064340          16677#
FDST1   064254          16649#
FDST2   064310          16665#
FDST3   064220          16635#
FEA     064336          16676#
FLAG    064450          16743#
FPSHI   064252          16648#
GNS   = ****** U          836   16366   16367   16368   16369
GNWHAM  064226          16638#
HI    = 000200           1225#  15485   15495
HT    = 000011            728#  16157   16198   16821   16822   16827   16837   16844   16846   16851   16872   17049   17055
                        17058   17063   17069   17073   17075   17084   17089   17092   17095
IBUF    063266           4581    4602   16393#
IDENT1  065141           4638   16854#
INCRSV  061202          15836   15851#
INIT    003262           1930#  15717   15850
INSTAB  063636          13858   16525#
IOTVEC= 000020            823#
I002    001642           1283    1289#
I004    001664           1312#
I005    001702           1327#
I006    001720           1342#
I007    001734           1356#
I010    001750           1371#
I011    001766           1386#
I012    002006           1402#
I013    002030           1418#
I014    002060           1435#
I015    002076           1449#
I016    002124           1470#
I017    002154           1489#
I020    002236           1521#
I021    002266           1543#
I022    002312           1563#
```

```
I023    002346          1581#
I024    002404          1605#
I025    002422          1618#
I026    002442          1631#
I027    002466          1646#
I030    002506          1660#
I031    002526          1673#
I032    002560          1689#
I033    002620          1711#
I034    002660          1732#
I035    002710          1752#
I036    002730          1766#
I037    002750          1780#
I040    002772          1796#
I041    003020          1812#
I042    003052          1834#
I043    003074          1851#
I044    003120          1867#
I045    003146          1882#
I046    003212          1906#
JAM     064444         16739#
JMP5    041420         11480   11495   11501#
JMP5A   041466         11510   11527#
LCDTA   064436         16733#
LCUA    064350         16683#
LF    = 000012           729#  16192   16198   16861   16865
LFGIN   064352         16684#
LJAM    064342         16680#
LKCSR = 177546          1264#  12305   12328   12346   12365   12395   12608   12639   12670   12701   12732   12767   12832
                       12879
LO    = 000100          1224#  15485   15495
LPBA    064346         16682#
LSERV   064344         16681#
LTAG    064356         16686#
LWHAM   064354         16685#
MBUF0   063312          2043   2045*   2059   2061*   2120   2138   2158   2177   2194   2199   2211   2216   2234
                        2239   2257    2262   2274    2278   2300   2311   2352   2358*  2371   2376*  2406   2410*
                        2423   2425    2654   2656    2673   2747   2781   2783   2807   2808   2831   2837*  2850
                        2856*  2913*   2914   2933*   2934   2946   2961   2967   2985   2991   3003   3008*  3021
                        3026*  3096    3115   3116    3314   3339   3364   3389   3414   3429   3440   3455   3466
                        3469   3481    3492   3495    3507   3635   3641*  3653   3659*  3671   3677*  3693   3699*
                        4107   4111*   4124*  4128    4485*  4486*  4987   5007   5022   5027   5040   5057   5083
                        5153   5173    5187   5192    5205   5309   5336   5363   5390   5573   5600   5628   5656
                        5684   5712    5740   5752    5770   5800   5802   5813   5831   5833   5857   5859   5870
                        5888   5890    5914   5926    5944   6591   6619   6652   6680   6694   6712   6739   6766
                        6793   6820    6847   6874    6900   6927   6954   6981   7008   7035   7062   7089   7116
                        7143   7170    7197   7224    7381   7383   7390   7408   7410   7430   7438   7456   7477
                        7479   7486    7508   7510    7530   7538   7556   7681*  7687   7730*  7736   7753   7774*
                        7792   7813    7834   7851    7863   7880   7892   7921   7950   7953   7979   7982   8008
                        8037   8066    8069   8088    8091   8110   8132   8263   8292   8325   8354   8516   8545
                        8574   8683    8713   8744    8758   8770   8796   8798   8822   8848   8850   8869   8892
                        8924   8955    8987   9020    9036   9057   9073   9094   9131   9168   9170   9205   9207
                        9246   9283    9320   9322    9352   9354   9384   9416   9446   9467   9488   9509   9511
                        9530   9554    9715   9736    9751   9763   9790   9794   9817   9844   9848   9866   9888
                        9910   9932    9954   9957    9976   9979  10245  10274  10303  10332  10361  10390  10418
                       10447  10476   10747  10776   10805  10835  10865  10899  10929  10959  10989  11037  11058
```

```
                    11080   11102   11105   11124   11146   11149   11168   11190   11211   11214   11233   11254   11257
                    12887*  12909*  12912   13863   13942*  13948   14342   14373   14393   14410*  14445   14446*  14465
                    14482*  14517   14549*  16399   16402#
MBUF1   063316       2157    2176    2714    2719*   2932    3313    3338    3363    3388    7846*   7850    7875*   7879
                     7904*   7908    7933*   7937    7962*   7966    7991*   7995    8020*   8024    8049*   8053    8657
                     8685    8715    9552   10559   10721   10807   10837   10867   10901   10931   10961   10991   12888*
                    12918   12925*  13157   13167   13196   13206   13864   13908*  13914   16400   16404#
MD      064440      16735#
MED   = 076600       1215#  13801   14908   14921   14952   14957   14960   14964   14998   15011   15040   15043   15051
                    15085   15088   15090   15093   15104   15117   15120   15124   15136   15139   15141   15162   15171
                    15186   15193   15205   15232   15234   15236   15241   15247   15253   15259   15265   15298   15303
                    15337   15352   15366   15383   15414   15455   15461   15470   15482   15503   15514   15528   15536
                    15542   15573   15589   15611   15626   15637   15655   15670   15750
MEDHLT  055632      14923   14924   14925   14927#
MEDT1   055642      14944#
MEDT10  056206      15045#
MEDT11  056260      15081#  15143
MEDT3   056042      14995#
MEDT4   056100      15009#
MED0    055612      14920#
MED1    055514      14901#
MEMERR= 177744       1218#  15306   15310   15494
NOPS    064404      16706#
NUA     064456      16749#
OBUF    063256       4582    4603   16388#
ONCE    063254       1927*   1932    4634    4636*  16385#
ORTAB   063474      14187   14192   16467#
PABORT= 000200       1223#  15447
PA1500  064162      16603#
PA1716  064160      16602#
PBA     064446      16741#
PIRQ  = 177772        735#
PIRQVE= 000240        829#
PRIFLG  063236       1930    4401*   4410*  16115*  16377#
PRINA   062202       4403   16115#
PR0   = 000000        752#
PR1   = 000040        753#
PR2   = 000100        754#
PR3   = 000140        755#
PR4   = 000200        756#
PR5   = 000240        757#
PR6   = 000300        758#
PR7   = 000340        759#
PS    = 177776        732#    733
PSW   = 177776        733#   1432    1446    1467    1485    1496    1505    1514    1563*   1577    1601    1605    1618
                     1631    1684    1685*   1706    1707*   1727    1728*   1791    1807    1997*   2012*   2026    4226*
                     4234    4238*   4257*   4262    4266*   4280*   4288    4292*   4299*   4306*   4341*   4349    4353*
                     4360*   4367*   4380*   4385    4388*   4639*   4681    4687    4701    4709    4723    4729    4743
                     4751   12110   12117*  12125   12147   12154*  12162   12211   12214   12271   12274   12371*  12402*
                    12417*  12428*  12434   12446   12449*  12461*  12467   12479   12482*  12507*  12513*  12530   12537*
                    12564*  12570*  12587   12594*  12613*  12628*  12644*  12659*  12675*  12690*  12706*  12721*  12737*
                    12752*  12772*  12788*  12804*  12820*  12840*  12850*  12886*  12907*  12943*  14682*  14711   14712
                    14902*  14920*  14944*  15774*
PSWHCL  064050      16597#
PTRP1   057712      15458   15469#
PWRUP   061054      15764   15773#
```

```
PWRVEC= 000024            824#  15728*  15729*  15738*  15744*  15759*  15760*
RCSR  = 177560           1260#   4492    4578
RDBR  = 177562           1261#
RDFLAG= 000144           1236#  15041   15091   15118
RDLCUA= 000103           1244#  15105
RDLDAT= 000106           1250#  15260   15529
RDLFGI= 000104           1246#  15384   15543   15656   15671
RDLJAM= 000100           1238#  15172   15338   15367   15415   15590   15627
RDLPBA= 000102           1242#  15187   15254   15353   15504
RDLSER= 000101           1240#  15194   15483
RDLTAG= 000107           1252#  15266   15515
RDLWHA= 000105           1248#
RDWHAM= 000022           1234#  15086   15137
RES     064457          16750#
RESTAR  061176          15839   15848   15850#  15853   15907   15914
RESVEC= 000010            819#  14905*  14918*
REV     064453          16746#
RSBERT  061150          15842#  15910
RSERR   061122           4456   13838   14918   15835#
RSMSG   065232          15841   16865#
RSVFLG  063246           4444*   4450*  15832*  16381#
RSVTST  061114           4441   15832#
RT1A    064372          16692#
RT1B    064376          16694#
RT2A    064374          16693#
RT2B    064400          16695#
RVECT   064322          16670#
RZERO   064320          16669#
R0A     064422          16723#
R0B     064426          16725#
R1A     064166          16622#
R1B     064256          16652#
R2A     064170          16623#
R2B     064260          16653#
R3A     064172          16624#
R3B     064262          16654#
R4A     064174          16625#
R4B     064264          16655#
R5A     064176          16626#
R5B     064266          16656#
R6A     064200          16627#
R6B     064270          16657#
R7A     064424          15009   16724#
R7B     064430          15021   16726#
SCOFLG  063244           4317*   4324*  15994*  16380#
SCOPEA  061612           4318   15994#
SELTST  063242          15940*  15941*  15942   16379#
SERV    064445          16740#
SOBERR  043012          11975   11984#
SOB1    042764          11970#  12053
SOB2    043000          11965   11968   11978#
SOB3    043156          11970   12051#
SOB4    043160          12049   12053#
SOB5    042772          11974#  11978
SREG    064454          16747#
STACK = 001000            723#   1431    1918    1934   14903   15773
```

```
START   001630            840    1274#  15775
STGBLK  064056          16600#
STKLMT= 177774            734#
SWB01 = 000071           1254#  15081  15099  15103
SWR     001040            901#   4631*  4632  15736  15752*  15838  15847  15906  15938  15940  15945  15963  15970
                        16033  16045  16048
SWREG   000176            838#
SW0   = 000001            787#
SW00  = 000001            777#    787
SW01  = 000002            776#    786
SW02  = 000004            775#    785
SW03  = 000010            774#    784
SW04  = 000020            773#    783
SW05  = 000040            772#    782
SW06  = 000100            771#    781
SW07  = 000200            770#    780
SW08  = 000400            769#    779
SW09  = 001000            768#    778
SW1   = 000002            786#
SW10  = 002000            767#  15938
SW11  = 004000            766#
SW12  = 010000            765#   4632  15838  15906
SW13  = 020000            764#
SW14  = 040000            763#
SW15  = 100000            762#
SW2   = 000004            785#
SW3   = 000010            784#
SW4   = 000020            783#
SW5   = 000040            782#
SW6   = 000100            781#
SW7   = 000200            780#
SW8   = 000400            779#
SW9   = 001000            778#
TABBEG  064052          16598#
TABEND  064054          16599#
TAG   = 000040           1226#  15485  15495
TBITVE= 000014            820#
TBL2    064166          14945  16619#
TBL3    064404          14995  16705#
TBL4    064422          16722#
TBL5    064436          15110  16731#
TBL6    064464          15111  16759#
TBL7    064536          15047  16787#
TBSER   061070          13846  15781#
TKVEC = 000060            827#
TLOC1   064046          15223  15238  15289  15296  15449  15459  15505  15509  15517  16596#
TLOC2   064164          16604#
TPVEC = 000064            828#
TRAPVE= 000034            826#
TRTVEC= 000014            821#
TSET    052422          13846#
TST0    003316           1942#  4640
TST1    003332           1947   1955#
TST10   003532           2050   2057#
TST100  007200           3228   3237#
TST101  007274           3265   3274#
```

TST102	007366	3302	3311#
TST103	007446	3329	3336#
TST104	007526	3354	3361#
TST105	007606	3379	3386#
TST106	007666	3404	3412#
TST107	007744	3430	3438#
TST11	003564	2066	2074#
TST110	010022	3456	3464#
TST111	010102	3482	3490#
TST112	010162	3508	3516#
TST113	010224	3531	3538#
TST114	010266	3553	3561#
TST115	010356	3597#	
TST116	010446	3633#	
TST117	010510	3644	3651#
TST12	003622	2088	2096#
TST120	010552	3662	3669#
TST121	010614	3680	3687#
TST122	010670	3702	3710#
TST123	010714	3718	3726#
TST124	010752	3746	3754#
TST125	010770	3760	3768#
TST126	011004	3774	3782#
TST127	011020	3788	3796#
TST13	003656	2110	2118#
TST130	011040	3803	3811#
TST131	011060	3818	3826#
TST132	011076	3833	3841#
TST133	011114	3847	3855#
TST134	011132	3862	3870#
TST135	011150	3877	3885#
TST136	011170	3892	3900#
TST137	011210	3907	3915#
TST14	003710	2128	2136#
TST140	011230	3922	3930#
TST141	011250	3937	3945#
TST142	011264	3951	3959#
TST143	011304	3966	3974#
TST144	011324	3981	3989#
TST145	011344	3996	4004#
TST146	011362	4011	4019#
TST147	011376	4025	4033#
TST15	003744	2146	2154#
TST150	011416	4040	4048#
TST151	011436	4055	4063#
TST152	011456	4070	4078#
TST153	011524	4096	4104#
TST154	011566	4114	4122#
TST155	011622	4131	4139#
TST156	011710	4167	4175#
TST157	011744	4188	4196#
TST16	004010	2165	2173#
TST160	012012	4212	4220#
TST161	012100	4243	4251#
TST162	012152	4264	4273#
TST163	012274	4304	4313#

TST164	012342	4334#	
TST165	012464	4365	4374#
TST166	012540	4397#	
TST167	012620	4411	4419#
TST17	004054	2184	2192#
TST170	012664	4431	4438#
TST171	012746	4462#	
TST172	013016	4475	4483#
TST173	013116	4511#	
TST174	013156	4521	4532#
TST175	013210	4541	4550#
TST176	013244	4559	4576#
TST177	013556	4648#	
TST2	003350	1961	1969#
TST20	004110	2202	2209#
TST200	013600	4655	4662#
TST201	013620	4669	4676#
TST202	013660	4689	4696#
TST203	013722	4711	4718#
TST204	013760	4731	4738#
TST205	014020	4753	4760#
TST206	014042	4767	4774#
TST207	014064	4782	4789#
TST21	004154	2224	2232#
TST210	014106	4797	4804#
TST211	014130	4812	4819#
TST212	014154	4827	4834#
TST213	014176	4841	4848#
TST214	014220	4856	4863#
TST215	014242	4871	4878#
TST216	014264	4886	4893#
TST217	014332	4912	4919#
TST22	004214	2247	2255#
TST220	014374	4933	4940#
TST221	014442	4959	4966#
TST222	014474	4976	4983#
TST223	014622	5028	5035#
TST224	014656	5045	5052#
TST225	014726	5071	5078#
TST226	014764	5089	5096#
TST227	015034	5115	5122#
TST23	004250	2264	2272#
TST230	015104	5141	5148#
TST231	015234	5193	5200#
TST232	015310	5220	5227#
TST233	015354	5246	5253#
TST234	015424	5272	5279#
TST235	015472	5298	5305#
TST236	015542	5325	5332#
TST237	015616	5352	5359#
TST24	004310	2286	2294#
TST240	015672	5379	5386#
TST241	015744	5406	5413#
TST242	016014	5432	5439#
TST243	016062	5458	5465#
TST244	016132	5484	5491#

TST245	016200	5510	5517#
TST246	016250	5536	5543#
TST247	016320	5562	5569#
TST25	004360	2312	2320#
TST250	016376	5589	5596#
TST251	016452	5616	5624#
TST252	016530	5644	5652#
TST253	016604	5672	5680#
TST254	016662	5700	5708#
TST255	016740	5728	5736#
TST256	017020	5758	5766#
TST257	017072	5784	5792#
TST26	004434	2341	2350#
TST260	017170	5819	5827#
TST261	017244	5845	5853#
TST262	017330	5876	5884#
TST263	017404	5902	5910#
TST264	017464	5932	5940#
TST265	017534	5957	5965#
TST266	017600	5984	5991#
TST267	017646	6011	6018#
TST27	004470	2361	2369#
TST270	017714	6037	6044#
TST271	017760	6062	6069#
TST272	020030	6088	6095#
TST273	020076	6114	6121#
TST274	020144	6140	6147#
TST275	020214	6166	6173#
TST276	020262	6192	6199#
TST277	020332	6218	6225#
TST3	003364	1974	1981#
TST30	004526	2379	2387#
TST300	020376	6243	6250#
TST301	020444	6269	6276#
TST302	020514	6295	6302#
TST303	020560	6321	6328#
TST304	020630	6347	6354#
TST305	020700	6372	6379#
TST306	020744	6398	6405#
TST307	021014	6424	6431#
TST31	004552	2395	2403#
TST310	021062	6450	6457#
TST311	021132	6476	6483#
TST312	021200	6502	6509#
TST313	021250	6528	6535#
TST314	021320	6554	6561#
TST315	021366	6580	6587#
TST316	021440	6607	6615#
TST317	021516	6636	6644#
TST32	004606	2413	2421#
TST320	021604	6668	6676#
TST321	021666	6700	6708#
TST322	021744	6727	6735#
TST323	022020	6754	6762#
TST324	022074	6781	6789#
TST325	022152	6808	6816#

TST326	022226	6835	6843#
TST327	022304	6862	6870#
TST33	004646	2432	2440#
TST330	022356	6888	6896#
TST331	022432	6915	6923#
TST332	022510	6942	6950#
TST333	022562	6969	6977#
TST334	022640	6996	7004#
TST335	022716	7023	7031#
TST336	022770	7050	7058#
TST337	023046	7077	7085#
TST34	004702	2453	2461#
TST340	023122	7104	7112#
TST341	023200	7131	7139#
TST342	023254	7158	7166#
TST343	023332	7185	7193#
TST344	023410	7212	7220#
TST345	023464	7239	7247#
TST346	023534	7266	7273#
TST347	023604	7292	7299#
TST35	004736	2474	2481#
TST350	023654	7318	7325#
TST351	023724	7344	7351#
TST352	023774	7370	7377#
TST353	024054	7396	7404#
TST354	024124	7418	7426#
TST355	024200	7444	7452#
TST356	024244	7465	7473#
TST357	024324	7492	7500#
TST36	004772	2494	2502#
TST360	024406	7518	7526#
TST361	024462	7544	7552#
TST362	024526	7565	7573#
TST363	024600	7593	7600#
TST364	024654	7620	7627#
TST365	024722	7644	7651#
TST366	024770	7668	7675#
TST367	025044	7693	7700#
TST37	005026	2515	2523#
TST370	025112	7717	7724#
TST371	025166	7742	7749#
TST372	025226	7761	7768#
TST373	025272	7781	7788#
TST374	025336	7801	7809#
TST375	025402	7822	7830#
TST376	025470	7852	7859#
TST377	025556	7881	7888#
TST4	003402	1987	1995#
TST40	005066	2537	2545#
TST400	025646	7910	7917#
TST401	025736	7939	7946#
TST402	026024	7968	7975#
TST403	026112	7997	8004#
TST404	026202	8026	8033#
TST405	026272	8055	8062#
TST406	026344	8076	8084#

TST407	026416	8098	8106#
TST41	005132	2560	2569#
TST410	026466	8120	8128#
TST411	026536	8142	8150#
TST412	026602	8169	8176#
TST413	026652	8196	8203#
TST414	026726	8223	8230#
TST415	027002	8250	8257#
TST416	027056	8278	8286#
TST417	027136	8307	8315#
TST42	005174	2583	2592#
TST420	027232	8340	8348#
TST421	027314	8369	8377#
TST422	027364	8397	8404#
TST423	027434	8424	8431#
TST424	027510	8451	8458#
TST425	027564	8478	8485#
TST426	027640	8505	8512#
TST427	027720	8533	8541#
TST43	005236	2606	2615#
TST430	030000	8562	8570#
TST431	030060	8590	8598#
TST432	030132	8618	8625#
TST433	030202	8645	8652#
TST434	030260	8672	8679#
TST435	030344	8701	8709#
TST436	030432	8732	8740#
TST437	030506	8759	8766#
TST44	005274	2636#	
TST440	030564	8785	8792#
TST441	030640	8811	8818#
TST442	030716	8837	8844#
TST443	030766	8857	8865#
TST444	031036	8878	8886#
TST445	031122	8909	8918#
TST446	031206	8941	8950#
TST447	031272	8973	8982#
TST45	005322	2644	2652#
TST450	031356	9005	9014#
TST451	031452	9042	9051#
TST452	031546	9079	9088#
TST453	031644	9116	9125#
TST454	031742	9153	9162#
TST455	032036	9190	9199#
TST456	032132	9227	9236#
TST457	032242	9268	9277#
TST46	005364	2663	2671#
TST460	032340	9305	9314#
TST461	032430	9337	9346#
TST462	032520	9369	9378#
TST463	032610	9401	9410#
TST464	032700	9433	9442#
TST465	032740	9455	9463#
TST466	033000	9476	9484#
TST467	033042	9497	9505#
TST47	005420	2681	2689#

TST470	033104	9518	9526#
TST471	033146	9539	9548#
TST472	033212	9561	9569#
TST473	033264	9589	9596#
TST474	033336	9617	9624#
TST475	033374	9637	9644#
TST476	033434	9656	9663#
TST477	033502	9680	9687#
TST5	003424	2002	2010#
TST50	005462	2704	2712#
TST500	033550	9704	9711#
TST501	033614	9724	9732#
TST502	033672	9752	9759#
TST503	033752	9778	9786#
TST504	034030	9805	9813#
TST505	034110	9832	9840#
TST506	034162	9854	9862#
TST507	034234	9876	9884#
TST51	005520	2721	2729#
TST510	034302	9898	9906#
TST511	034350	9920	9928#
TST512	034420	9942	9950#
TST513	034470	9964	9972#
TST514	034542	9986	9994#
TST515	034616	10014	10021#
TST516	034676	10045	10052#
TST517	034752	10072	10079#
TST52	005544	2744#	
TST520	035020	10099	10106#
TST521	035076	10126	10134#
TST522	035150	10154	10161#
TST523	035222	10181	10188#
TST524	035274	10208	10215#
TST525	035346	10234	10241#
TST526	035430	10262	10270#
TST527	035504	10291	10299#
TST53	005600	2754	2762#
TST530	035566	10320	10328#
TST531	035642	10349	10357#
TST532	035724	10378	10386#
TST533	036006	10407	10414#
TST534	036070	10435	10443#
TST535	036150	10464	10472#
TST536	036230	10492	10500#
TST537	036304	10520	10527#
TST54	005632	2771	2779#
TST540	036354	10547	10554#
TST541	036434	10575	10582#
TST542	036504	10602	10609#
TST543	036556	10629	10636#
TST544	036630	10656	10662#
TST545	036702	10682	10689#
TST546	036754	10709	10716#
TST547	037032	10736	10743#
TST55	005700	2796	2804#
TST550	037114	10764	10772#

TST551	037172	10793	10801#			
TST552	037260	10823	10831#			
TST553	037340	10853	10861#			
TST554	037426	10883	10891#			
TST555	037526	10917	10925#			
TST556	037612	10947	10955#			
TST557	037676	10977	10985#			
TST56	005750	2821	2829#			
TST560	037762	11006	11014#			
TST561	040020	11026	11033#			
TST562	040064	11046	11054#			
TST563	040132	11068	11076#			
TST564	040202	11090	11098#			
TST565	040252	11112	11120#			
TST566	040322	11134	11142#			
TST567	040374	11156	11164#			
TST57	006010	2840	2848#			
TST570	040446	11178	11186#			
TST571	040510	11199	11207#			
TST572	040560	11221	11229#			
TST573	040622	11242	11250#			
TST574	040672	11264	11272#			
TST575	040732	11282	11287	11294#		
TST576	040772	11304	11309	11316#		
TST577	041042	11326	11336	11343#		
TST6	003444	2016	2024#			
TST60	006062	2865	2872#			
TST600	041102	11353	11358	11365#		
TST601	041160	11375	11385	11388	11397#	
TST602	041226	11407	11412	11415	11423#	
TST603	041304	11433	11437	11447	11454#	
TST604	041344	11464	11469	11476#		
TST605	041422	11486	11496	11499	11506#	
TST606	041470	11516	11521	11524	11532#	
TST607	041536	11542	11547	11550	11558#	
TST61	006124	2887	2895#			
TST610	041604	11568	11573	11576	11584#	
TST611	041660	11594	11597	11602	11605	11614#
TST612	041734	11624	11627	11632	11635	11644#
TST613	042012	11660	11672#			
TST614	042100	11703#				
TST615	042162	11732#				
TST616	042232	11757#				
TST617	042310	11773	11786#			
TST62	006156	2904	2911#			
TST620	042374	11802	11819#			
TST621	042456	11838	11851#			
TST622	042542	11867	11884#			
TST623	042622	11900	11912#			
TST624	042710	11929	11946#			
TST625	042740	11954	11962#			
TST626	043014	11980	11983	11989#		
TST627	043050	12001	12008#			
TST63	006216	2921	2929#			
TST630	043104	12020	12027#			
TST631	043140	12039	12046#			

```
TST632  043174          12058   12065#
TST633  043276          12095   12106#
TST634  043402          12132   12143#
TST635  043510          12169   12180#
TST636  043666          12196   12204   12216   12225   12240#
TST637  044044          12256   12264   12276   12285   12300#
TST64   006274           2947    2955#
TST640  044126          12324#
TST641  044162          12334   12342#
TST642  044220          12352   12360#
TST643  044332          12390#
TST644  044450          12422#
TST645  044566          12455#
TST646  044702          12488#
TST647  045106          12545#
TST65   006352           2975    2983#
TST650  045310          12602#
TST651  045426          12633#
TST652  045544          12664#
TST653  045662          12695#
TST654  046000          12726#
TST655  046116          12757#
TST656  046250          12793#
TST657  046370          12828#
TST66   006406           2993    3001#
TST660  046530          12873#
TST661  047030          12956#
TST662  047076          12976#
TST663  047232          13016#
TST664  047346          13054#
TST665  047430          13079#
TST666  047512          13104#
TST667  047574          13129#
TST67   006444           3011    3019#
TST670  047660          13153#
TST671  047766          13184   13192#
TST672  050074          13222   13230#
TST673  050234          13261   13282#
TST674  050416          13330   13342#
TST675  050600          13390   13402#
TST676  050674          13432#
TST677  051000          13464#
TST7    003476           2034    2041#
TST70   006504           3029    3037#
TST700  051104          13496#
TST701  051164          13519#
TST702  051244          13542#
TST703  051324          13564#
TST704  051424          13597#
TST705  051530          13615   13632#
TST706  051636          13650   13667#
TST707  051744          13685   13702#
TST71   006534           3046    3054#
TST710  052032          13728#
TST711  052120          13755#
TST712  052206          13782#
```

```
TST713  052304          13812#
TST714  052436          13855#
TST715  052526          13882#
TST716  052556          13897#
TST717  052644          13922#
TST72   006604           3078#
TST720  052700          13938#
TST721  052742          13955#
TST722  053010          14004#
TST723  053074          14064#
TST724  053160          14124#
TST725  053244          14184#
TST726  053330          14230#
TST727  053404          14248   14274#
TST73   006634           3086    3094#
TST730  053460          14292   14304#
TST731  053544          14327   14334#
TST732  053634          14358   14365#
TST733  053732          14394   14401#
TST734  054036          14430   14437#
TST735  054142          14466   14473#
TST736  054246          14502   14509#
TST737  054340          14533   14540#
TST74   006672           3104    3112#
TST740  054422          14557   14564#
TST741  054506          14586   14593#
TST742  054574          14615   14622#
TST743  054656          14644   14651#
TST744  054730          14666   14677#
TST745  055046          14709   14719#
TST746  055116          14737   14743#
TST747  055166          14762   14768#
TST75   006734           3123    3131#
TST750  055236          14786   14792#
TST751  055324          14814   14820#
TST752  055406          14843   14849#
TST753  055506          14875   14898#
TST754  055634          14941#
TST755  056034          14992#
TST756  056164          15036#
TST757  056252          15078#
TST76   007012           3155    3163#
TST760  056554          15156#
TST761  056732          15219#
TST762  057142          15286#
TST763  057274          15327#
TST764  057502          15402#
TST765  057600          15443#
TST766  060160          15568#
TST767  060402          15643#
TST77   007104           3191    3200#
TYPE  = 104401           4406    4637  15701  15704  15707  15761  15840  15844  15908  16036  16071  16088  16090
                        16093   16095  16099  16106  16162  16259  16366#
TYPOC = 104402          15703   15706  15843  16079  16103  16367#
TYPON = 104404          16369#
TYPOS = 104403          16368#
```

```
UBREAK= 177770          1217#  1935*  14901*  15081*  15115*  15145*
UCNSCT  064474         16765#
UCOUNT  064522         16776#
UCUA    064506         16770#
UDCS0   064530         16779#
UDCS1   064534         16781#
UDREG   064514         16773#
UFLAG   064510         16771#
UINIT   064532         16780#
UJAM    064500         16767#
ULCDTA  064464         16761#
UM    = 140000          1216#
UMD     064470         16763#
UNUA    064524         16777#
UPBA    064504         16769#
URES    064526         16778#
UREV    064516         16774#
UR6A    064216         16634#
UR6B    064306         16664#
USERV   064502         16768#
USREG   064520         16775#
VADR    064156         16601#
WCNSSW= 000226          1233#  15751
WCSADR  064316         16668#
WCSA.0  064222         16636#
WCSA.1  064224         16637#
WCSB.0  064312         16666#
WCSB.1  064314         16667#
WRFLAG= 000344          1237#  15044  15094  15121  15140
WRLCUA= 000303          1245#
WRLDAT= 000306          1251#  15235
WRLFGI= 000304          1247#
WRLJAM= 000300          1239#
WRLPBA= 000302          1243#  15233
WRLSER= 000301          1241#
WRLTAG= 000307          1253#  15237
WRLWHA= 000305          1249#
WRWHAM= 000222          1235#  15089  15142  15163  15206  15456  15537  15574  15612  15638
WWP   = 000100          1221#  15225  15291  15293  15451  15453
XCSR  = 177564          1262#   4513   4534   4552  12427  12450*  12460  12483*  12494  12551  12799  12880
XDBR  = 177566          1263#
$APTHD  000700           851    857#
$ASTAT= ****** U       16311  16326
$ATYC   062744         16282  16284#
$ATY1   062720         16280#
$ATY3   062726         16147  16281#
$ATY4   062736         16041  16283#
$AUTOB  001034           898#
$BDADR  001022           893#
$BDDAT  001026           895#
$CHARC  062466         16164*  16174*  16181  16190*  16195#
$CKSWR= ****** U       16372
$CMTAG  001000           881#
$CM1  = 000006           913#   914#   915#   916#   917#   918#   919#
$CM2  = 000014           913#   914#   915#   916#   917#   918#   919#
$CM3  = 000006           911#   913
```

```
$CM4   = 000005            919#    920#    921#    922#    923#    924#
$CPUOP   001146            948#
$CRLF    001115            927#  15707   15845   16036   16059   16071   16090   16095   16099   16163   16198
$DEVCT   001130            939#
$DOAGN   060654          15697   15709   15715#
$ENDAD   060644            870   15711#
$ENDCT   060600          15699#
$ENULL   060660          15718#
$ENV     001140            944#  16038   16142   16289   16313
$ENVM    001141            945#   4629   16144   16149   16291
$EOP     060550          15690#
$EOPCT   060572          15696#  15700
$ERFLG   001003            884#  15923   15959   15961   15967*  15993   16026*  16059
$ERMAX   001015            890#  15961   15984*  15993
$ERROR   061620           4623   15650   16013#
$ERRPC   001016            891#  16030*  16031*  16032   16059   16077   17099   17101   17103   17105   17106   17108   17109
                         17111   17115   17127   17129   17132
$ERRTB   001150            971#  16085
$ERRTY   062046          16035   16070#
$ERTTL   001012            888#   1928*  15705   15846*  16029*  16059
$ESCAP   001112            925#  15934*  15936*  15983*  16051   16053   16059
$ETABL   001140            943#
$ETEND   001150            863     955#
$FATAL   001122            936#  16317*
$FFLG    063164          16280*  16283*  16311   16320*  16328#
$FILLC   001056            909#  16167   16198
$FILLS   001055            908#  16198
$GDADR   001020            892#
$GDDAT   001024            894#
$GET42   060634          15708#
$GTSWR=  ****** U          16371
$HD    = 000000            708
$HIBTS   000700            858#
$ICNT    001004            885#  15974*  15975   15977*  15992
$ILLUP   061036          15728   15744   15766#
$INTAG   001035            899#
$ITEMB   001014            889#  16032*  16040   16059   16074
$LF      001116            928#  16059   16198
$LFLG    063163          16321*  16327#
$LPADR   001006            886#   4640*  15965*  15981*  15986   15992
$LPERR   001010            887#  11649*  11677*  11712*  11737*  11762*  11791*  11824*  11856*  11889*  11917*  12070*  12112*
                         12149*  12306*  12367*  12396*  12493*  12550*  12607*  12638*  12669*  12700*  12731*  12766*  12798*
                         12834*  12878*  12961*  13134*  13501*  13524*  13547*  13569*  13709*  13724*  13735*  13750*  13762*
                         13777*  13789*  13807*  13819*  13837*  13861*  13877*  13906*  13960*  14008*  14030*  14068*  14090*
                         14128*  14150*  14188*  14210*  14236*  14255*  14280*  14299*  15112*  15143*  15965   15982*  15992
                         16050
$MAIL    001120            859     863     934#  15980   16038   16142
$MBADR   000702            859#
$MFLG    063162          16281*  16287   16322*  16326#
$MSGAD   001134            941#  16297*  16300
$MSGLG   001136            942#  16302*
$MSGTY   001120            935#  16295   16303*  16315   16319*
$MXCNT   061610          15978   15992#
$NULL    001054            907#  16169   16198
$NWTST=  000001           1939#   1952#   1966#   1978#   1992#   2007#   2021#   2038#   2054#   2071#   2093#   2115#   2133#
                          2151#   2170#   2189#   2206#   2229#   2252#   2269#   2291#   2317#   2347#   2366#   2384#   2400#
```

```
                2418#  2437#  2458#  2478#  2499#  2520#  2542#  2566#  2589#  2612#  2633#  2649#  2668#
2686#  2709#  2726#  2741#  2759#  2776#  2801#  2826#  2845#  2869#  2892#  2908#  2926#
2952#  2980#  2998#  3016#  3034#  3051#  3075#  3091#  3109#  3128#  3160#  3197#  3234#
3271#  3308#  3333#  3358#  3383#  3409#  3435#  3461#  3487#  3513#  3535#  3558#  3594#
3630#  3648#  3666#  3684#  3707#  3723#  3751#  3765#  3779#  3793#  3808#  3823#  3838#
3852#  3867#  3882#  3897#  3912#  3927#  3942#  3956#  3971#  3986#  4001#  4016#  4030#
4045#  4060#  4075#  4101#  4119#  4136#  4172#  4193#  4217#  4248#  4270#  4310#  4331#
4371#  4394#  4416#  4435#  4459#  4480#  4508#  4529#  4547#  4565#  4567    4645#  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#  5621#  5649#  5677#  5705#  5733#  5763#  5789#  5824#  5850#  5881#  5907#  5937#
5962#  5988#  6015#  6041#  6066#  6092#  6118#  6144#  6170#  6196#  6222#  6247#  6273#
6299#  6325#  6351#  6376#  6402#  6428#  6454#  6480#  6506#  6532#  6558#  6584#  6612#
6641#  6673#  6705#  6732#  6759#  6786#  6813#  6840#  6867#  6893#  6920#  6947#  6974#
7001#  7028#  7055#  7082#  7109#  7136#  7163#  7190#  7217#  7244#  7270#  7296#  7322#
7348#  7374#  7401#  7423#  7449#  7470#  7497#  7523#  7549#  7570#  7597#  7624#  7648#
7672#  7697#  7721#  7746#  7765#  7785#  7806#  7827#  7856#  7885#  7914#  7943#  7972#
8001#  8030#  8059#  8081#  8103#  8125#  8147#  8173#  8200#  8227#  8254#  8283#  8312#
8345#  8374#  8401#  8428#  8455#  8482#  8509#  8538#  8567#  8595#  8622#  8649#  8676#
8706#  8737#  8763#  8789#  8815#  8841#  8862#  8883#  8915#  8947#  8979#  9011#  9048#
9085#  9122#  9159#  9196#  9233#  9274#  9311#  9343#  9375#  9407#  9439#  9460#  9481#
9502#  9523#  9545#  9566#  9593#  9621#  9641#  9660#  9684#  9708#  9729#  9756#  9783#
9810#  9837#  9859#  9881#  9903#  9925#  9947#  9969#  9991#  10018#  10049#  10076#  10103#
10131#  10158#  10185#  10212#  10238#  10267#  10296#  10325#  10354#  10383#  10411#  10440#  10469#
10497#  10524#  10551#  10579#  10606#  10633#  10659#  10686#  10713#  10740#  10769#  10798#  10828#
10858#  10888#  10922#  10952#  10982#  11011#  11030#  11051#  11073#  11095#  11117#  11139#  11161#
11183#  11204#  11226#  11247#  11269#  11291#  11313#  11340#  11362#  11394#  11420#  11451#  11473#
11503#  11529#  11555#  11581#  11611#  11641#  11669#  11700#  11729#  11754#  11783#  11816#  11848#
11881#  11909#  11943#  11959#  11986#  12005#  12024#  12043#  12062#  12103#  12140#  12177#  12237#
12297#  12321#  12339#  12357#  12387#  12419#  12452#  12485#  12542#  12599#  12630#  12661#  12692#
12723#  12754#  12790#  12822#  12824  12866#  12868  12953#  12973#  13013#  13051#  13076#  13101#
13126#  13150#  13189#  13227#  13279#  13339#  13399#  13429#  13461#  13493#  13516#  13539#  13561#
13592#  13594  13629#  13664#  13699#  13725#  13752#  13779#  13809#  13849#  13851  13879#  13894#
13919#  13935#  13952#  13972#  13974  14032#  14034  14092#  14094  14152#  14154  14212#  14214
14257#  14259  14301#  14331#  14362#  14398#  14434#  14470#  14506#  14537#  14561#  14590#  14619#
14648#  14670#  14672  14716#  14740#  14765#  14789#  14817#  14846#  14891#  14893  14929#  14931
14980#  14982  15026#  15028  15061#  15063  15148#  15150  15209#  15211  15279#  15281  15316#
15318  15392#  15394  15431#  15433  15553#  15555  15640#

$OCNT   062714        16231*  16260*  16273#
$OMODE  062716        16226*  16230*  16235  16238*  16249*  16275#
$OVER   061560        15943  15946  15966  15976  15985#
$PASS   001126          938#   1929*  15693*  15694*  15702  15718  15972  15993
$PASTM  000706          861#
$POWER  061044        15762  15769#
$PWRAD  061032        15764#
$PWRDN  060664         4627  15728#  15759
$PWRMG  061026        15762#
$PWRUP  060736        15738  15744#
$QUES   001114          926#  16059  16198
$RDCHR= ****** U      16372
$RDDEC= ****** U      16372
$RDLIN= ****** U      16372
$RDOCT= ****** U      16372
$REGAD  001060          911#  16015  17113  17127  17129
$REGO   001062          913#  15448*  15459*  15476  15546*  17101  17106  17108  17109  17115  17127  17129  17132
```

```
$REG1   001064          914#  17106  17109  17111  17113  17127  17129
$REG2   001066          915#  17109  17111  17113  17126
$REG3   001070          916#  17109  17111  17113  17125  17129
$REG4   001072          917#  17113  17124  17129
$REG5   001074          918#   1936* 12192* 12200* 12212* 12230* 12252* 12260* 12272* 12290* 12991* 12999* 13006*
                      13030* 13038* 13045* 13069* 13094* 13119* 13295* 13305* 13315* 13325* 13355* 13365* 13375*
                      13385* 13413* 13419* 13423* 13445* 13454* 13477* 13486* 16055* 17127
$RTNAD  060656        15717#
$R2A  = ****** U      16372
$SAVRE= ****** U      16372
$SAVR6  061042        15737* 15745  15746* 15747* 15768#
$SCOPE  061260         4621  15665  15931#
$SETUP= 000000        15692  15932  16014  16048  16055
$SVLAD  061524        15954  15979#
$SVPC = 000714          868#   873
$SWR  = 165000          698#   708   712   713   714   715   716   717   718   719   924   925   926
                       1944  1957  1971  1983  1997  2012  2026  2043  2059  2076  2098  2120  2138
                       2156  2175  2194  2211  2234  2257  2274  2296  2322  2352  2371  2389  2405
                       2423  2442  2463  2483  2504  2525  2547  2571  2594  2617  2638  2654  2673
                       2691  2714  2731  2746  2764  2781  2806  2831  2850  2874  2897  2913  2931
                       2957  2985  3003  3021  3039  3056  3080  3096  3114  3133  3165  3202  3239
                       3276  3313  3338  3363  3388  3414  3440  3466  3492  3518  3540  3563  3599
                       3635  3653  3671  3689  3712  3728  3756  3770  3784  3798  3813  3828  3843
                       3857  3872  3887  3902  3917  3932  3947  3961  3976  3991  4006  4021  4035
                       4050  4065  4080  4106  4124  4141  4177  4198  4222  4253  4275  4315  4336
                       4376  4399  4421  4440  4464  4485  4513  4534  4552  4578  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  5888  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  8986  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 11965 11993 12012 12031 12049 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
                      13705 13731 13758 13785 13815 13857 13886 13901 13926 13942 13959 14007 14067
                      14127 14187 14233 14277 14307 14337 14368 14404 14440 14476 14512 14543 14567
                      14596 14625 14654 14681 14723 14747 14772 14795 14823 14852 14901 14944 14995
                      15039 15081 15159 15222 15289 15330 15405 15446 15571 15646 15687 15692 15710
                      15716 15718 15765 15924 15925 15926 15927 15928 15945 15957 15959 15960 15961
```

```
                    15968   15969   15970   15982   15985   15992   16006   16007   16008   16009   16029   16033   16045
                    16048   16059
$SWREG  001142        946#    4631
$SWRMK= 000000      15928
$TESTN  001124        937#    4642*   15932   15980*  15990*  17132
$TIMES  001110        924#    4641*   12426*  12459*  15692*  15968*  15975   15978*  15992
$TKB    001046        904#
$TKS    001044        903#
$TMP0   001076        919#   14307*  14337*  14368*  14404*  14440*  14476*  14512*  14543*  14567*  14596*  14625*  14654*
                    14795*  14823*  14856*  14954*  14963   15098*  15128*  17099   17115
$TMP1   001100        920#   14955*  14968*  14974   15015*  15055*  15099*  15129*  17099   17101   17103
$TMP2   001102        921#   14946*  14956   14966   14973*  14975   15016*  15056*  17101   17103
$TMP3   001104        922#   14962*  14966   15017*  17103
$TMP4   001106        923#   14951*  14969   14972*
$TN   = 000770        698#     708   1939    1943    1944#   1947    1952    1956    1957#   1961    1966    1970    1971#
                     1974    1978    1982    1983#   1987    1992    1996    1997#   2002    2007    2011    2012#   2016
                     2021    2025    2026#   2034    2038    2042    2043#   2050    2054    2058    2059#   2066    2071
                     2075    2076#   2088    2093    2097    2098#   2110    2115    2119    2120#   2128    2133    2137
                     2138#   2146    2151    2155    2156#   2165    2170    2174    2175#   2184    2189    2193    2194#
                     2202    2206    2210    2211#   2224    2229    2233    2234#   2247    2252    2256    2257#   2264
                     2269    2273    2274#   2286    2291    2295    2296#   2312    2317    2321    2322#   2341    2347
                     2351    2352#   2361    2366    2370    2371#   2379    2384    2388    2389#   2395    2400    2404
                     2405#   2413    2418    2422    2423#   2432    2437    2441    2442#   2453    2458    2462    2463#
                     2474    2478    2482    2483#   2494    2499    2503    2504#   2515    2520    2524    2525#   2537
                     2542    2546    2547#   2560    2566    2570    2571#   2583    2589    2593    2594#   2606    2612
                     2616    2617#   2633    2637    2638#   2644    2649    2653    2654#   2663    2668    2672    2673#
                     2681    2686    2690    2691#   2704    2709    2713    2714#   2721    2726    2730    2731#   2741
                     2745    2746#   2754    2759    2763    2764#   2771    2776    2780    2781#   2796    2801    2805
                     2806#   2821    2826    2830    2831#   2840    2845    2849    2850#   2865    2869    2873    2874#
                     2887    2892    2896    2897#   2904    2908    2912    2913#   2921    2926    2930    2931#   2947
                     2952    2956    2957#   2975    2980    2984    2985#   2993    2998    3002    3003#   3011    3016
                     3020    3021#   3029    3034    3038    3039#   3046    3051    3055    3056#   3075    3079    3080#
                     3086    3091    3095    3096#   3104    3109    3113    3114#   3123    3128    3132    3133#   3155
                     3160    3164    3165#   3191    3197    3201    3202#   3228    3234    3238    3239#   3265    3271
                     3275    3276#   3302    3308    3312    3313#   3329    3333    3337    3338#   3354    3358    3362
                     3363#   3379    3383    3387    3388#   3404    3409    3413    3414#   3430    3435    3439    3440#
                     3456    3461    3465    3466#   3482    3487    3491    3492#   3508    3513    3517    3518#   3531
                     3535    3539    3540#   3553    3558    3562    3563#   3594    3598    3599#   3630    3634    3635#
                     3644    3648    3652    3653#   3662    3666    3670    3671#   3680    3684    3688    3689#   3702
                     3707    3711    3712#   3718    3723    3727    3728#   3746    3751    3755    3756#   3760    3765
                     3769    3770#   3774    3779    3783    3784#   3788    3793    3797    3798#   3803    3808    3812
                     3813#   3818    3823    3827    3828#   3833    3838    3842    3843#   3847    3852    3856    3857#
                     3862    3867    3871    3872#   3877    3882    3886    3887#   3892    3897    3901    3902#   3907
                     3912    3916    3917#   3922    3927    3931    3932#   3937    3942    3946    3947#   3951    3956
                     3960    3961#   3966    3971    3975    3976#   3981    3986    3990    3991#   3996    4001    4005
                     4006#   4011    4016    4020    4021#   4025    4030    4034    4035#   4040    4045    4049    4050#
                     4055    4060    4064    4065#   4070    4075    4079    4080#   4096    4101    4105    4106#   4114
                     4119    4123    4124#   4131    4136    4140    4141#   4167    4172    4176    4177#   4188    4193
                     4197    4198#   4212    4217    4221    4222#   4243    4248    4252    4253#   4264    4270    4274
                     4275#   4304    4310    4314    4315#   4331    4335    4336#   4365    4371    4375    4376#   4394
                     4398    4399#   4411    4416    4420    4421#   4431    4435    4439    4440#   4459    4463    4464#
                     4475    4480    4484    4485#   4508    4512    4513#   4521    4529    4533    4534#   4541    4547
                     4551    4552#   4559    4565    4577    4578#   4645    4650    4652#   4655    4659    4664    4666#
                     4669    4673    4678    4680#   4689    4693    4698    4700#   4711    4715    4720    4722#   4731
                     4735    4740    4742#   4753    4757    4762    4764#   4767    4771    4776    4778#   4782    4786
                     4791    4793#   4797    4801    4806    4808#   4812    4816    4821    4823#   4827    4831    4836
```

```
4838#  4841   4845   4850   4852#  4856   4860   4865   4867#  4871   4875   4880   4882#
4886   4890   4895   4897#  4912   4916   4921   4923#  4933   4937   4942   4944#  4959
4963   4968   4970#  4976   4980   4985   4987#  5028   5032   5037   5039#  5045   5049
5054   5056#  5071   5075   5080   5082#  5089   5093   5098   5100#  5115   5119   5124
5126#  5141   5145   5150   5152#  5193   5197   5202   5204#  5220   5224   5229   5231#
5246   5250   5255   5257#  5272   5276   5281   5283#  5298   5302   5307   5309#  5325
5329   5334   5336#  5352   5356   5361   5363#  5379   5383   5388   5390#  5406   5410
5415   5417#  5432   5436   5441   5443#  5458   5462   5467   5469#  5484   5488   5493
5495#  5510   5514   5519   5521#  5536   5540   5545   5547#  5562   5566   5571   5573#
5589   5593   5598   5600#  5616   5621   5626   5628#  5644   5649   5654   5656#  5672
5677   5682   5684#  5700   5705   5710   5712#  5728   5733   5738   5740#  5758   5763
5768   5770#  5784   5789   5794   5796#  5819   5824   5829   5831#  5845   5850   5855
5857#  5876   5881   5886   5888#  5902   5907   5912   5914#  5932   5937   5942   5944#
5957   5962   5967   5969#  5984   5988   5993   5995#  6011   6015   6020   6022#  6037
6041   6046   6048#  6062   6066   6071   6073#  6088   6092   6097   6099#  6114   6118
6123   6125#  6140   6144   6149   6151#  6166   6170   6175   6177#  6192   6196   6201
6203#  6218   6222   6227   6229#  6243   6247   6252   6254#  6269   6273   6278   6280#
6295   6299   6304   6306#  6321   6325   6330   6332#  6347   6351   6356   6358#  6372
6376   6381   6383#  6398   6402   6407   6409#  6424   6428   6433   6435#  6450   6454
6459   6461#  6476   6480   6485   6487#  6502   6506   6511   6513#  6528   6532   6537
6539#  6554   6558   6563   6565#  6580   6584   6589   6591#  6607   6612   6617   6619#
6636   6641   6646   6648#  6668   6673   6678   6680#  6700   6705   6710   6712#  6727
6732   6737   6739#  6754   6759   6764   6766#  6781   6786   6791   6793#  6808   6813
6818   6820#  6835   6840   6845   6847#  6862   6867   6872   6874#  6888   6893   6898
6900#  6915   6920   6925   6927#  6942   6947   6952   6954#  6969   6974   6979   6981#
6996   7001   7006   7008#  7023   7028   7033   7035#  7050   7055   7060   7062#  7077
7082   7087   7089#  7104   7109   7114   7116#  7131   7136   7141   7143#  7158   7163
7168   7170#  7185   7190   7195   7197#  7212   7217   7222   7224#  7239   7244   7249
7251#  7266   7270   7275   7277#  7292   7296   7301   7303#  7318   7322   7327   7329#
7344   7348   7353   7355#  7370   7374   7379   7381#  7396   7401   7406   7408#  7418
7423   7428   7430#  7444   7449   7454   7456#  7465   7470   7475   7477#  7492   7497
7502   7504#  7518   7523   7528   7530#  7544   7549   7554   7556#  7565   7570   7575
7577#  7593   7597   7602   7604#  7620   7624   7629   7631#  7644   7648   7653   7655#
7668   7672   7677   7679#  7693   7697   7702   7704#  7717   7721   7726   7728#  7742
7746   7751   7753#  7761   7765   7770   7772#  7781   7785   7790   7792#  7801   7806
7811   7813#  7822   7827   7832   7834#  7852   7856   7861   7863#  7881   7885   7890
7892#  7910   7914   7919   7921#  7939   7943   7948   7950#  7968   7972   7977   7979#
7997   8001   8006   8008#  8026   8030   8035   8037#  8055   8059   8064   8066#  8076
8081   8086   8088#  8098   8103   8108   8110#  8120   8125   8130   8132#  8142   8147
8152   8154#  8169   8173   8178   8180#  8196   8200   8205   8207#  8223   8227   8232
8234#  8250   8254   8259   8261#  8278   8283   8288   8290#  8307   8312   8317   8319#
8340   8345   8350   8352#  8369   8374   8379   8381#  8397   8401   8406   8408#  8424
8428   8433   8435#  8451   8455   8460   8462#  8478   8482   8487   8489#  8505   8509
8514   8516#  8533   8538   8543   8545#  8562   8567   8572   8574#  8590   8595   8600
8602#  8618   8622   8627   8629#  8645   8649   8654   8656#  8672   8676   8681   8683#
8701   8706   8711   8713#  8732   8737   8742   8744#  8759   8763   8768   8770#  8785
8789   8794   8796#  8811   8815   8820   8822#  8837   8841   8846   8848#  8857   8862
8867   8869#  8878   8883   8888   8890#  8909   8915   8920   8922#  8941   8947   8952
8954#  8973   8979   8984   8986#  9005   9011   9016   9018#  9042   9048   9053   9055#
9079   9085   9090   9092#  9116   9122   9127   9129#  9153   9159   9164   9166#  9190
9196   9201   9203#  9227   9233   9238   9240#  9268   9274   9279   9281#  9305   9311
9316   9318#  9337   9343   9348   9350#  9369   9375   9380   9382#  9401   9407   9412
9414#  9433   9439   9444   9446#  9455   9460   9465   9467#  9476   9481   9486   9488#
9497   9502   9507   9509#  9518   9523   9528   9530#  9539   9545   9550   9552#  9561
9566   9571   9573#  9589   9593   9598   9600#  9617   9621   9626   9628#  9637   9641
9646   9648#  9656   9660   9665   9667#  9680   9684   9689   9691#  9704   9708   9713
```

```
                9715#   9724    9729    9734    9736#   9752    9756    9761    9763#   9778    9783    9788    9790#
                9805    9810    9815    9817#   9832    9837    9842    9844#   9854    9859    9864    9866#   9876
                9881    9886    9888#   9898    9903    9908    9910#   9920    9925    9930    9932#   9942    9947
                9952    9954#   9964    9969    9974    9976#   9986    9991    9996    9998#   10014   10018   10023
               10025#  10045   10049   10054   10056#  10072   10076   10081   10083#  10099   10103   10108   10110#
               10126   10131   10136   10138#  10154   10158   10163   10165#  10181   10185   10190   10192#  10208
               10212   10217   10219#  10234   10238   10243   10245#  10262   10267   10272   10274#  10291   10296
               10301   10303#  10320   10325   10330   10332#  10349   10354   10359   10361#  10378   10383   10388
               10390#  10407   10411   10416   10418#  10435   10440   10445   10447#  10464   10469   10474   10476#
               10492   10497   10502   10504#  10520   10524   10529   10531#  10547   10551   10556   10558#  10575
               10579   10584   10586#  10602   10606   10611   10613#  10629   10633   10638   10640#  10656   10659
               10664   10666#  10682   10686   10691   10693#  10709   10713   10718   10720#  10736   10740   10745
               10747#  10764   10769   10774   10776#  10793   10798   10803   10805#  10823   10828   10833   10835#
               10853   10858   10863   10865#  10883   10888   10893   10895#  10917   10922   10927   10929#  10947
               10952   10957   10959#  10977   10982   10987   10989#  11006   11011   11016   11018#  11026   11030
               11035   11037#  11046   11051   11056   11058#  11068   11073   11078   11080#  11090   11095   11100
               11102#  11112   11117   11122   11124#  11134   11139   11144   11146#  11156   11161   11166   11168#
               11178   11183   11188   11190#  11199   11204   11209   11211#  11221   11226   11231   11233#  11242
               11247   11252   11254#  11264   11269   11274   11276#  11282   11287   11291   11296   11298#  11304
               11309   11313   11318   11320#  11326   11336   11340   11345   11347#  11353   11358   11362   11367
               11369#  11375   11385   11388   11394   11399   11401#  11407   11412   11415   11420   11425   11427#
               11433   11437   11447   11451   11456   11458#  11464   11469   11473   11478   11480#  11486   11496
               11499   11503   11508   11510#  11516   11521   11524   11529   11534   11536#  11542   11547   11550
               11555   11560   11562#  11568   11573   11576   11581   11586   11588#  11594   11597   11602   11605
               11611   11616   11618#  11624   11627   11632   11635   11641   11646   11648#  11660   11669   11674
               11676#  11700   11705   11707#  11729   11734   11736#  11754   11759   11761#  11773   11783   11788
               11790#  11802   11816   11821   11823#  11838   11848   11853   11855#  11867   11881   11886   11888#
               11900   11909   11914   11916#  11929   11943   11948   11950#  11954   11959   11964   11965#  11980
               11983   11986   11991   11993#  12001   12005   12010   12012#  12020   12024   12029   12031#  12039
               12043   12048   12049#  12058   12062   12067   12069#  12095   12103   12108   12110#  12132   12140
               12145   12147#  12169   12177   12182   12184#  12196   12204   12216   12225   12237   12242   12244#
               12256   12264   12276   12285   12297   12302   12304#  12321   12326   12328#  12334   12339   12344
               12346#  12352   12357   12362   12364#  12387   12392   12394#  12419   12424   12426#  12452   12457
               12459#  12485   12490   12492#  12542   12547   12549#  12599   12604   12606#  12630   12635   12637#
               12661   12666   12668#  12692   12697   12699#  12723   12728   12730#  12754   12759   12761#  12790
               12795   12797#  12822   12830   12832#  12866   12875   12877#  12953   12958   12960#  12973   12978
               12980#  13013   13018   13020#  13051   13056   13058#  13076   13081   13083#  13101   13106   13108#
               13126   13131   13133#  13150   13155   13157#  13184   13189   13194   13196#  13222   13227   13232
               13234#  13261   13279   13284   13286#  13330   13339   13344   13346#  13390   13399   13404   13406#
               13429   13434   13436#  13461   13466   13468#  13493   13498   13500#  13516   13521   13523#  13539
               13544   13546#  13561   13566   13568#  13592   13599   13601#  13615   13629   13634   13636#  13650
               13664   13669   13671#  13685   13699   13704   13705#  13725   13730   13731#  13752   13757   13758#
               13779   13784   13785#  13809   13814   13815#  13849   13856   13857#  13879   13884   13886#  13894
               13899   13901#  13919   13924   13926#  13935   13940   13942#  13952   13957   13959#  13972   14006
               14007#  14032   14066   14067#  14092   14126   14127#  14152   14186   14187#  14212   14232   14233#
               14248   14257   14276   14277#  14292   14301   14306   14307#  14327   14331   14336   14337#  14358
               14362   14367   14368#  14394   14398   14403   14404#  14430   14434   14439   14440#  14466   14470
               14475   14476#  14502   14506   14511   14512#  14533   14537   14542   14543#  14557   14561   14566
               14567#  14586   14590   14595   14596#  14615   14619   14624   14625#  14644   14648   14653   14654#
               14666   14670   14679   14681#  14709   14716   14721   14723#  14737   14740   14745   14747#  14762
               14765   14770   14772#  14786   14789   14794   14795#  14814   14817   14822   14823#  14843   14846
               14851   14852#  14875   14891   14899   14901#  14929   14942   14944#  14980   14993   14995#  15026
               15037   15039#  15061   15079   15081#  15148   15157   15159#  15209   15220   15222#  15279   15287
               15289#  15316   15328   15330#  15392   15403   15405#  15431   15444   15446#  15553   15569   15571#
               15640   15644   15646#  15677
$TPB    001052  906#   16187*  16198
```

```
$TPFLG  001057            910#  16136   16198
$TPS    001050            905#  16185   16198
$TRAP   063166           4408    4625  16343#
$TRAP2  063210          16354#  16365
$TRP    = 000005        16358#  16367#  16368#  16369#  16370#
$TRPAD  063222          16348   16365#
$TSTM   000704            860#
$TSTNM  001002            883#  15923   15937*  15979*  15980   15985   15987   15993   16028   16059
$TYPBN= ****** U        16370
$TYPDS= ****** U        16370
$TYPE   062210          16136#  16308   16358   16366
$TYPEC  062422          16166   16173   16180   16185#  16186
$TYPEX  062470          16191   16193   16196#
$TYPOC  062516          16229#  16367
$TYPON  062532          16228   16231#  16369
$TYPOS  062472          16224#  16368
$UNIT   001132            940#
$UNITM  000710            862#
$USWR   001144            947#
$XTSTR  061350          15948#
$$GET4= 000000          15710#
$OFILL  062715          16225*  16229*  16239   16274#
$40CAT= ****** U        15945   16035
.       = 070034          832#    836#    841#    847     848#    850#    852#    868     869#    871#    873#    880#    929
                         2298    2739    2959    3691    4200    4925    5798    6650    7506    8321    9242   10027   10897
                        11709   12408   12619   12650   12681   12712   12743   12763   12778   12810   13903   14854   14926
                        15718   15719#  15740   15767   15992   15993   16059   16109#  16198   16329#  16393#  16577   16578
                        16600#  17098#  17120#
.$ASTA= ****** U        16281   16284
.$X     = 000700          847#    852
```

```
COMMEN      1#    830#
ENDCOM      1#    830#
ENDERR  15999#  16055
ENDPAS  15682#  15701
ENDSCO  15917#  15987
ERROR     724#   4428   4657   4671   4691   4707   4713   4733   4749   4755   4769   4784   4799   4814   4829
         4843   4858   4873   4888   4909   4914   4935   4955   4961   4978   5000   5005   5020   5025   5030
         5047   5068   5073   5091   5112   5117   5138   5143   5165   5170   5185   5190   5195   5217   5222
         5243   5248   5269   5274   5295   5300   5322   5327   5349   5354   5376   5381   5403   5408   5429
         5434   5455   5460   5481   5486   5507   5512   5533   5538   5559   5564   5586   5591   5613   5619
         5641   5647   5669   5675   5697   5703   5725   5731   5750   5755   5761   5781   5787   5811   5816
         5822   5842   5848   5868   5873   5879   5899   5905   5924   5929   5935   5954   5960   5981   5986
         6008   6013   6034   6039   6059   6064   6085   6090   6111   6116   6137   6142   6163   6168   6189
         6194   6215   6220   6240   6245   6266   6271   6292   6297   6318   6323   6344   6349   6370   6374
         6395   6400   6421   6426   6447   6452   6473   6478   6499   6504   6525   6530   6551   6556   6577
         6582   6604   6610   6633   6639   6665   6671   6692   6697   6703   6725   6730   6752   6757   6779
         6784   6806   6811   6833   6838   6860   6865   6886   6891   6913   6918   6940   6945   6967   6972
         6994   6999   7021   7026   7048   7053   7075   7080   7102   7107   7129   7134   7156   7161   7183
         7188   7210   7215   7237   7242   7263   7268   7289   7294   7315   7320   7341   7346   7367   7372
         7393   7399   7421   7441   7447   7468   7489   7495   7521   7541   7547   7568   7590   7595   7617
         7622   7641   7646   7665   7670   7690   7695   7714   7719   7739   7744   7763   7783   7804   7825
         7848   7854   7877   7883   7906   7912   7935   7941   7964   7970   7993   7999   8022   8028   8051
         8057   8079   8101   8123   8145   8166   8171   8193   8198   8220   8225   8247   8252   8275   8281
         8304   8310   8337   8343   8366   8372   8394   8399   8421   8426   8448   8453   8475   8480   8502
         8507   8530   8536   8559   8565   8587   8593   8615   8620   8642   8647   8669   8674   8698   8704
         8729   8735   8756   8761   8782   8787   8808   8813   8834   8839   8860   8881   8906   8913   8938
         8945   8970   8977   9002   9009   9034   9039   9046   9071   9076   9083   9108   9113   9120   9145
         9150   9157   9182   9187   9194   9219   9224   9231   9260   9265   9272   9297   9302   9309   9334
         9341   9366   9373   9398   9405   9430   9437   9458   9479   9500   9521   9542   9564   9586   9591
         9614   9619   9639   9658   9677   9682   9701   9706   9727   9749   9754   9775   9781   9802   9808
         9829   9835   9857   9879   9901   9923   9945   9967   9989  10011  10016  10042  10047  10069  10074
        10096  10101  10123  10129  10151  10156  10178  10183  10205  10210  10231  10236  10259  10265  10288
        10294  10317  10323  10346  10352  10375  10381  10404  10410  10432  10438  10461  10467  10489  10495
        10517  10522  10544  10549  10572  10577  10599  10604  10626  10631  10653  10658  10679  10684  10706
        10711  10733  10738  10761  10767  10790  10796  10820  10826  10850  10856  10880  10886  10914  10920
        10944  10950  10974  10980  11003  11009  11028  11049  11071  11093  11115  11137  11159  11181  11202
        11224  11245  11267  11281  11289  11303  11311  11325  11333  11338  11352  11360  11374  11382  11387
        11390  11406  11414  11418  11432  11436  11444  11449  11463  11471  11485  11493  11498  11501  11515
        11523  11527  11541  11549  11552  11567  11575  11578  11593  11596  11604  11609  11623  11626  11634
        11639  11656  11666  11686  11691  11696  11719  11726  11744  11750  11769  11779  11798  11808  11812
        11831  11834  11844  11863  11873  11877  11896  11906  11924  11935  11939  11953  11982  11984  12003
        12022  12041  12060  12079  12087  12092  12099  12122  12129  12136  12159  12166  12173  12194  12202
        12215  12223  12233  12254  12262  12275  12283  12293  12316  12337  12355  12380  12411  12439  12447
        12472  12480  12515  12522  12531  12534  12572  12579  12588  12591  12622  12653  12684  12715  12746
        12782  12814  12856  12860  12915  12922  12929  12935  12940  12968  12993  13001  13008  13032  13040
        13047  13071  13096  13121  13144  13165  13173  13181  13204  13212  13219  13242  13250  13258  13275
        13297  13307  13317  13327  13357  13367  13377  13387  13415  13425  13447  13456  13479  13488  13511
        13534  13557  13579  13586  13614  13624  13649  13659  13684  13694  13717  13743  13770  13798  13827
        13874  13892  13916  13933  13950  13970  14025  14085  14145  14205  14246  14290  14322  14329  14353
        14360  14384  14391  14396  14420  14427  14432  14456  14463  14468  14492  14499  14504  14528  14535
        14559  14581  14588  14610  14617  14639  14646  14668  14714  14734  14738  14759  14763  14783  14787
        14809  14815  14838  14844  14870  14876  14910  14914  14927  14971  15018  15057  15100  15109  15130
        15167  15177  15182  15201  15244  15273  15300  15311  15343  15349  15357  15371  15379  15387  15409
        15420  15426  15463  15479  15488  15499  15510  15525  15533  15547  15595  15606  15629  15659  15674
        15935
ESCAPE      1#    830#
```

```
GETPRI      1#    830#
GETSWR      1#    830#
MSG      4565#   4567  12822#  12824  12866#  12868  13592#  13594  13849#  13851  13972#  13974  14032#  14034  14092#
        14094  14152#  14154  14212#  14214  14257#  14259  14670#  14672  15147#  15150  15208#  15211  15278#  15281
        15315#  15318  15391#  15394  15430#  15433  15552#  15555
MSGJ    14891#  14893
MSGM1   14928#  14931
MSGM10  15025#  15028
MSGM11  15060#  15063
MSGM3   14980#  14982
MULT        1#    830#
NEWTST      1#    830#  1939  1952  1966  1978  1992  2007  2021  2038  2054  2071  2093  2115  2133
         2151  2170  2189  2206  2229  2252  2269  2291  2317  2347  2366  2384  2400  2418  2437
         2458  2478  2499  2520  2542  2566  2589  2612  2633  2649  2668  2686  2709  2726  2741
         2759  2776  2801  2826  2845  2869  2892  2908  2926  2952  2980  2998  3016  3034  3051
         3075  3091  3109  3128  3160  3197  3234  3271  3308  3333  3358  3383  3409  3435  3461
         3487  3513  3535  3558  3594  3630  3648  3666  3684  3707  3723  3751  3765  3779  3793
         3808  3823  3838  3852  3867  3882  3897  3912  3927  3942  3956  3971  3986  4001  4016
         4030  4045  4060  4075  4101  4119  4136  4172  4193  4217  4248  4270  4310  4331  4371
         4394  4416  4435  4459  4480  4508  4529  4547  4565  4645  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  5621  5649  5677  5705  5733  5763  5789  5824  5850
         5881  5907  5937  5962  5988  6015  6041  6066  6092  6118  6144  6170  6196  6222  6247
         6273  6299  6325  6351  6376  6402  6428  6454  6480  6506  6532  6558  6584  6612  6641
         6673  6705  6732  6759  6786  6813  6840  6867  6893  6920  6947  6974  7001  7028  7055
         7082  7109  7136  7163  7190  7217  7244  7270  7296  7322  7348  7374  7401  7423  7449
         7470  7497  7523  7549  7570  7597  7624  7648  7672  7697  7721  7746  7765  7785  7806
         7827  7856  7885  7914  7943  7972  8001  8030  8059  8081  8103  8125  8147  8173  8200
         8227  8254  8283  8312  8345  8374  8401  8428  8455  8482  8509  8538  8567  8595  8622
         8649  8676  8706  8737  8763  8789  8815  8841  8862  8883  8915  8947  8979  9011  9048
         9085  9122  9159  9196  9233  9274  9311  9343  9375  9407  9439  9460  9481  9502  9523
         9545  9566  9593  9621  9641  9660  9684  9708  9729  9756  9783  9810  9837  9859  9881
         9903  9925  9947  9969  9991 10018 10049 10076 10103 10131 10158 10185 10212 10238 10267
        10296 10325 10354 10383 10411 10440 10469 10497 10524 10551 10579 10606 10633 10659 10686
        10713 10740 10769 10798 10828 10858 10888 10922 10952 10982 11011 11030 11051 11073 11095
        11117 11139 11161 11183 11204 11226 11247 11269 11291 11313 11340 11362 11394 11420 11451
        11473 11503 11529 11555 11581 11611 11641 11669 11700 11729 11754 11783 11816 11848 11881
        11909 11943 11959 11986 12005 12024 12043 12062 12103 12140 12177 12237 12297 12321 12339
        12357 12387 12419 12452 12485 12542 12599 12630 12661 12692 12723 12754 12790 12822 12866
        12953 12973 13013 13051 13076 13101 13126 13150 13189 13227 13279 13339 13399 13429 13461
        13493 13516 13539 13561 13592 13629 13664 13699 13725 13752 13779 13809 13849 13879 13894
        13919 13935 13952 13972 14032 14092 14152 14212 14257 14301 14331 14362 14398 14434 14470
        14506 14537 14561 14590 14619 14648 14670 14716 14740 14765 14789 14817 14846 14891 14929
        14980 15026 15061 15148 15209 15279 15316 15392 15431 15553 15640
NOINST    1266#  11962  12046  13702  13728  13755  13782  13812  14004  14064  14124  14184  14230  14274  14304
        14334  14365  14401  14437  14473  14509  14540  14564  14593  14622  14651  14792  14820  14849
NOSCOP    1266#  1942  1955  1969  1981  1995  2010  2024  2041  2057  2074  2096  2118  2136  2154
         2173  2192  2209  2232  2255  2272  2294  2320  2350  2369  2387  2403  2421  2440  2461
         2481  2502  2523  2545  2569  2592  2615  2636  2652  2671  2689  2712  2729  2744  2762
         2779  2804  2829  2848  2872  2895  2911  2929  2955  2983  3001  3019  3037  3054  3078
         3094  3112  3131  3163  3200  3237  3274  3311  3336  3361  3386  3412  3438  3464  3490
         3516  3538  3561  3597  3633  3651  3669  3687  3710  3726  3754  3768  3782  3796  3811
         3826  3841  3855  3870  3885  3900  3915  3930  3945  3959  3974  3989  4004  4019  4033
         4048  4063  4078  4104  4122  4139  4175  4196  4220  4251  4273  4313  4334  4374  4397
         4419  4438  4462  4483  4511  4532  4550  4576 13855
```

```
POP          1#    830#  15752  15753  16323  16324
PREERR   15998#  16014
PRENEW    1266#  14898  14941  14992  15036  15078  15156  15219  15286  15327  15402  15443  15568  15643
PRESCO   15916#  15932
PUSH         1#    830#  15730  15736  16284  16286  16307
REPORT       1#    830#
SCOPE      725#   4322   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   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  12829  12874  12957  12977  13017  13055  13080  13105  13130
         13154  13193  13231  13283  13343  13403  13433  13465  13497  13520  13543  13565  13598  13633  13668
         13703  13729  13756  13783  13813  13840  13883  13898  13923  13939  13956  14005  14065  14125  14185
         14231  14275  14305  14335  14366  14402  14438  14474  14510  14541  14565  14594  14623  14652  14678
         14720  14744  14769  14793  14821  14850  14900  14943  14994  15038  15080  15158  15221  15288  15329
         15404  15445  15570  15645  15691
SETPRI       1#    830#
SETTRA   16358#  16367  16368  16369
SETUP        1#    830#
SKIP         1#    830#  1946   1961   1973   1987   2002   2015   2034   2050   2066   2088   2110   2128   2146
          2165   2184   2202   2224   2247   2264   2286   2312   2341   2361   2379   2395   2413   2432   2453
          2474   2494   2515   2537   2560   2583   2606   2644   2663   2681   2704   2721   2754   2771   2796
          2821   2840   2865   2887   2904   2921   2947   2975   2993   3011   3029   3046   3086   3104   3123
          3155   3191   3228   3265   3302   3329   3354   3379   3404   3430   3456   3482   3508   3531   3553
          3644   3662   3680   3702   3718   3745   3760   3773   3787   3803   3818   3832   3847   3861   3876
          3892   3907   3922   3937   3950   3966   3981   3996   4010   4024   4040   4055   4070   4096   4114
          4131   4167   4188   4212   4243   4264   4304   4365   4411   4431   4475   4521   4541   4559   4655
          4668   4689   4711   4731   4753   4767   4781   4796   4811   4827   4841   4855   4870   4885   4912
          4933   4959   4976   5028   5045   5071   5089   5115   5141   5193   5220   5246   5272   5298   5325
          5352   5379   5406   5432   5458   5484   5510   5536   5562   5589   5616   5644   5672   5700   5728
          5758   5784   5819   5845   5876   5902   5932   5957   5984   6011   6037   6062   6088   6114   6140
          6166   6192   6218   6243   6269   6295   6321   6347   6372   6398   6424   6450   6476   6502   6528
          6554   6580   6607   6636   6668   6700   6727   6754   6781   6808   6835   6862   6888   6915   6942
          6969   6996   7023   7050   7077   7104   7131   7158   7185   7212   7239   7266   7292   7318   7344
          7370   7396   7418   7444   7465   7492   7518   7544   7565   7593   7620   7644   7668   7693   7717
          7742   7761   7781   7801   7822   7852   7881   7910   7939   7968   7997   8026   8055   8076   8098
          8120   8142   8169   8196   8223   8250   8278   8307   8340   8369   8397   8424   8451   8478   8505
          8533   8562   8590   8618   8645   8672   8701   8732   8759   8785   8811   8837   8857   8878   8909
          8941   8973   9005   9042   9079   9116   9153   9190   9227   9268   9305   9337   9369   9401   9433
          9455   9476   9497   9518   9539   9561   9589   9617   9637   9656   9680   9704   9724   9752   9778
```

	9805	9832	9854	9876	9898	9920	9942	9964	9986	10014	10045	10072	10099	10126	10154
	10181	10208	10234	10262	10291	10320	10349	10378	10407	10435	10464	10492	10520	10547	10575
	10602	10629	10656	10682	10709	10736	10764	10793	10823	10853	10883	10917	10947	10977	11006
	11026	11046	11068	11090	11112	11134	11156	11178	11199	11221	11242	11264	11282	11287	11304
	11309	11326	11336	11353	11358	11375	11385	11388	11407	11412	11415	11433	11437	11447	11464
	11469	11486	11496	11499	11516	11521	11524	11542	11547	11550	11568	11573	11576	11594	11597
	11602	11605	11624	11627	11632	11635	11660	11773	11802	11838	11867	11900	11929	11954	11980
	11983	12001	12020	12039	12058	12095	12132	12169	12196	12204	12216	12225	12256	12264	12276
	12285	12334	12352	13184	13222	13261	13330	13390	13615	13650	13685	14248	14292	14327	14358
	14394	14430	14466	14502	14533	14557	14586	14615	14644	14666	14709	14737	14762	14786	14814
	14843	14875													
SLASH	1#	830#													
SPACE	830#														
STARS	1#	830#	844	846	853	866	876	929	932	1939	1941	1952	1954	1966	1968
	1978	1980	1992	1994	2007	2009	2021	2023	2038	2040	2054	2056	2071	2073	2093
	2095	2115	2117	2133	2135	2151	2153	2170	2172	2189	2191	2206	2208	2229	2231
	2252	2254	2269	2271	2291	2293	2317	2319	2347	2349	2366	2368	2384	2386	2400
	2402	2418	2420	2437	2439	2458	2460	2478	2480	2499	2501	2520	2522	2542	2544
	2566	2568	2589	2591	2612	2614	2633	2635	2649	2651	2668	2670	2686	2688	2709
	2711	2726	2728	2741	2743	2759	2761	2776	2778	2801	2803	2826	2828	2845	2847
	2869	2871	2892	2894	2908	2910	2926	2928	2952	2954	2980	2982	2998	3000	3016
	3018	3034	3036	3051	3053	3075	3077	3091	3093	3109	3111	3128	3130	3160	3162
	3197	3199	3234	3236	3271	3273	3308	3310	3333	3335	3358	3360	3383	3385	3409
	3411	3435	3437	3461	3463	3487	3489	3513	3515	3535	3537	3558	3560	3594	3596
	3630	3632	3648	3650	3666	3668	3684	3686	3707	3709	3723	3725	3751	3753	3765
	3767	3779	3781	3793	3795	3808	3810	3823	3825	3838	3840	3852	3854	3867	3869
	3882	3884	3897	3899	3912	3914	3927	3929	3942	3944	3956	3958	3971	3973	3986
	3988	4001	4003	4016	4018	4030	4032	4045	4047	4060	4062	4075	4077	4101	4103
	4119	4121	4136	4138	4172	4174	4193	4195	4217	4219	4248	4250	4270	4272	4310
	4312	4331	4333	4371	4373	4394	4396	4416	4418	4435	4437	4459	4461	4480	4482
	4508	4510	4529	4531	4547	4549	4565	4575	4645	4647	4659	4661	4673	4675	4693
	4695	4715	4717	4735	4737	4757	4759	4771	4773	4786	4788	4801	4803	4816	4818
	4831	4833	4845	4847	4860	4862	4875	4877	4890	4892	4916	4918	4937	4939	4963
	4965	4980	4982	5032	5034	5049	5051	5075	5077	5093	5095	5119	5121	5145	5147
	5197	5199	5224	5226	5250	5252	5276	5278	5302	5304	5329	5331	5356	5358	5383
	5385	5410	5412	5436	5438	5462	5464	5488	5490	5514	5516	5540	5542	5566	5568
	5593	5595	5621	5623	5649	5651	5677	5679	5705	5707	5733	5735	5763	5765	5789
	5791	5824	5826	5850	5852	5881	5883	5907	5909	5937	5939	5962	5964	5988	5990
	6015	6017	6041	6043	6066	6068	6092	6094	6118	6120	6144	6146	6170	6172	6196
	6198	6222	6224	6247	6249	6273	6275	6299	6301	6325	6327	6351	6353	6376	6378
	6402	6404	6428	6430	6454	6456	6480	6482	6506	6508	6532	6534	6558	6560	6584
	6586	6612	6614	6641	6643	6673	6675	6705	6707	6732	6734	6759	6761	6786	6788
	6813	6815	6840	6842	6867	6869	6893	6895	6920	6922	6947	6949	6974	6976	7001
	7003	7028	7030	7055	7057	7082	7084	7109	7111	7136	7138	7163	7165	7190	7192
	7217	7219	7244	7246	7270	7272	7296	7298	7322	7324	7348	7350	7374	7376	7401
	7403	7423	7425	7449	7451	7470	7472	7497	7499	7523	7525	7549	7551	7570	7572
	7597	7599	7624	7626	7648	7650	7672	7674	7697	7699	7721	7723	7746	7748	7765
	7767	7785	7787	7806	7808	7827	7829	7856	7858	7885	7887	7914	7916	7943	7945
	7972	7974	8001	8003	8030	8032	8059	8061	8081	8083	8103	8105	8125	8127	8147
	8149	8173	8175	8200	8202	8227	8229	8254	8256	8283	8285	8312	8314	8345	8347
	8374	8376	8401	8403	8428	8430	8455	8457	8482	8484	8509	8511	8538	8540	8567
	8569	8595	8597	8622	8624	8649	8651	8676	8678	8706	8708	8737	8739	8763	8765
	8789	8791	8815	8817	8841	8843	8862	8864	8883	8885	8915	8917	8947	8949	8979
	8981	9011	9013	9048	9050	9085	9087	9122	9124	9159	9161	9196	9198	9233	9235
	9274	9276	9311	9313	9343	9345	9375	9377	9407	9409	9439	9441	9460	9462	9481
	9483	9502	9504	9523	9525	9545	9547	9566	9568	9593	9595	9621	9623	9641	9643

	9660	9662	9684	9686	9708	9710	9729	9731	9756	9758	9783	9785	9810	9812	9837
	9839	9859	9861	9881	9883	9903	9905	9925	9927	9947	9949	9969	9971	9991	9993
	10018	10020	10049	10051	10076	10078	10103	10105	10131	10133	10158	10160	10185	10187	10212
	10214	10238	10240	10267	10269	10296	10298	10325	10327	10354	10356	10383	10385	10411	10413
	10440	10442	10469	10471	10497	10499	10524	10526	10551	10553	10579	10581	10606	10608	10633
	10635	10659	10661	10686	10688	10713	10715	10740	10742	10769	10771	10798	10800	10828	10830
	10858	10860	10888	10890	10922	10924	10952	10954	10982	10984	11011	11013	11030	11032	11051
	11053	11073	11075	11095	11097	11117	11119	11139	11141	11161	11163	11183	11185	11204	11206
	11226	11228	11247	11249	11269	11271	11291	11293	11313	11315	11340	11342	11362	11364	11394
	11396	11420	11422	11451	11453	11473	11475	11503	11505	11529	11531	11555	11557	11581	11583
	11611	11613	11641	11643	11669	11671	11700	11702	11729	11731	11754	11756	11783	11785	11816
	11818	11848	11850	11881	11883	11909	11911	11943	11945	11959	11961	11986	11988	12005	12007
	12024	12026	12043	12045	12062	12064	12103	12105	12140	12142	12177	12179	12237	12239	12297
	12299	12321	12323	12339	12341	12357	12359	12387	12389	12419	12421	12452	12454	12485	12487
	12542	12544	12599	12601	12630	12632	12661	12663	12692	12694	12723	12725	12754	12756	12790
	12792	12822	12827	12866	12872	12953	12955	12973	12975	13013	13015	13051	13053	13076	13078
	13101	13103	13126	13128	13150	13152	13189	13191	13227	13229	13279	13281	13339	13341	13399
	13401	13429	13431	13461	13463	13493	13495	13516	13518	13539	13541	13561	13563	13592	13596
	13629	13631	13664	13666	13699	13701	13725	13727	13752	13754	13779	13781	13809	13811	13849
	13854	13879	13881	13894	13896	13919	13921	13935	13937	13952	13954	13972	14003	14032	14063
	14092	14123	14152	14183	14212	14229	14257	14273	14301	14303	14331	14333	14362	14364	14398
	14400	14434	14436	14470	14472	14506	14508	14537	14539	14561	14563	14590	14592	14619	14621
	14648	14650	14670	14676	14716	14718	14740	14742	14765	14767	14789	14791	14817	14819	14846
	14848	14891	14897	14929	14940	14980	14991	15026	15035	15061	15077	15148	15155	15209	15218
	15279	15285	15316	15326	15392	15401	15431	15442	15553	15567	15640	15642	15685	15726	15742
	15786	15920	16002	16065	16121	16201	16279	16337							
SWRSU	1#	830#													
TESTNO	1266#	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	4577	4650	4664	4678	4698	4720	4740	4762
	4776	4791	4806	4821	4836	4850	4865	4880	4895	4921	4942	4968	4985	5037	5054
	5080	5098	5124	5150	5202	5229	5255	5281	5307	5334	5361	5388	5415	5441	5467
	5493	5519	5545	5571	5598	5626	5654	5682	5710	5738	5768	5794	5829	5855	5886
	5912	5942	5967	5993	6020	6046	6071	6097	6123	6149	6175	6201	6227	6252	6278
	6304	6330	6356	6381	6407	6433	6459	6485	6511	6537	6563	6589	6617	6646	6678
	6710	6737	6764	6791	6818	6845	6872	6898	6925	6952	6979	7006	7033	7060	7087
	7114	7141	7168	7195	7222	7249	7275	7301	7327	7353	7379	7406	7428	7454	7475
	7502	7528	7554	7575	7602	7629	7653	7677	7702	7726	7751	7770	7790	7811	7832
	7861	7890	7919	7948	7977	8006	8035	8064	8086	8108	8130	8152	8178	8205	8232
	8259	8288	8317	8350	8379	8406	8433	8460	8487	8514	8543	8572	8600	8627	8654
	8681	8711	8742	8768	8794	8820	8846	8867	8888	8920	8952	8984	9016	9053	9090
	9127	9164	9201	9238	9279	9316	9348	9380	9412	9444	9465	9486	9507	9528	9550
	9571	9598	9626	9646	9665	9689	9713	9734	9761	9788	9815	9842	9864	9886	9908
	9930	9952	9974	9996	10023	10054	10081	10108	10136	10163	10190	10217	10243	10272	10301
	10330	10359	10388	10416	10445	10474	10502	10529	10556	10584	10611	10638	10664	10691	10718
	10745	10774	10803	10833	10863	10893	10927	10957	10987	11016	11035	11056	11078	11100	11122
	11144	11166	11188	11209	11231	11252	11274	11296	11318	11345	11367	11399	11425	11456	11478
	11508	11534	11560	11586	11616	11646	11674	11705	11734	11759	11788	11821	11853	11886	11914
	11948	11964	11991	12010	12029	12048	12067	12108	12145	12182	12242	12302	12326	12344	12362
	12392	12424	12457	12490	12547	12604	12635	12666	12697	12728	12759	12795	12830	12875	12958

```
            12978   13018   13056   13081   13106   13131   13155   13194   13232   13284   13344   13404   13434   13466   13498
            13521   13544   13566   13599   13634   13669   13704   13730   13757   13784   13814   13856   13884   13899   13924
            13940   13957   14006   14066   14126   14186   14232   14276   14306   14336   14367   14403   14439   14475   14511
            14542   14566   14595   14624   14653   14679   14721   14745   14770   14794   14822   14851   14899   14942   14993
            15037   15079   15157   15220   15287   15328   15403   15444   15569   15644
TRMTRP      16358#
TYPBIN          1#    830#
TYPDEC          1#    830#
TYPNAM          1#    830#
TYPNUM          1#    830#
TYPOCS          1#    830#
TYPOCT          1#    830#  16077   16101
TYPTXT          1#    830#
UPCODE      15724#  15749
YESCOP       1266#   4648    4662    4676    4696    4718    4738    4760    4774    4789    4804    4819    4834    4848    4863
             4878    4893    4919    4940    4966    4983    5035    5052    5078    5096    5122    5148    5200    5227    5253
             5279    5305    5332    5359    5386    5413    5439    5465    5491    5517    5543    5569    5596    5624    5652
             5680    5708    5736    5766    5792    5827    5853    5884    5910    5940    5965    5991    6018    6044    6069
             6095    6121    6147    6173    6199    6225    6250    6276    6302    6328    6354    6379    6405    6431    6457
             6483    6509    6535    6561    6587    6615    6644    6676    6708    6735    6762    6789    6816    6843    6870
             6896    6923    6950    6977    7004    7031    7058    7085    7112    7139    7166    7193    7220    7247    7273
             7299    7325    7351    7377    7404    7426    7452    7473    7500    7526    7552    7573    7600    7627    7651
             7675    7700    7724    7749    7768    7788    7809    7830    7859    7888    7917    7946    7975    8004    8033
             8062    8084    8106    8128    8150    8176    8203    8230    8257    8286    8315    8348    8377    8404    8431
             8458    8485    8512    8541    8570    8598    8625    8652    8679    8709    8740    8766    8792    8818    8844
             8865    8886    8918    8950    8982    9014    9051    9088    9125    9162    9199    9236    9277    9314    9346
             9378    9410    9442    9463    9484    9505    9526    9548    9569    9596    9624    9644    9663    9687    9711
             9732    9759    9786    9813    9840    9862    9884    9906    9928    9950    9972    9994   10021   10052   10079
            10106   10134   10161   10188   10215   10241   10270   10299   10328   10357   10386   10414   10443   10472   10500
            10527   10554   10582   10609   10636   10662   10689   10716   10743   10772   10801   10831   10861   10891   10925
            10955   10985   11014   11033   11054   11076   11098   11120   11142   11164   11186   11207   11229   11250   11272
            11294   11316   11343   11365   11397   11423   11454   11476   11506   11532   11558   11584   11614   11644   11672
            11703   11732   11757   11786   11819   11851   11884   11912   11946   11989   12008   12027   12065   12106   12143
            12180   12240   12300   12324   12342   12360   12390   12422   12455   12488   12545   12602   12633   12664   12695
            12726   12757   12793   12828   12873   12956   12976   13016   13054   13079   13104   13129   13153   13192   13230
            13282   13342   13402   13432   13464   13496   13519   13542   13564   13597   13632   13667   13882   13897   13922
            13938   13955   14677   14719   14743   14768
$$CMRE       874#    913     914     915     916     917     918
$$CMTM       874#    919     920     921     922     923
$$ESCA          1#    830#
$$NEWT          1#    830#  1939    1952    1966    1978    1992    2007    2021    2038    2054    2071    2093    2115    2133
             2151    2170    2189    2206    2229    2252    2269    2291    2317    2347    2366    2384    2400    2418    2437
             2458    2478    2499    2520    2542    2566    2589    2612    2633    2649    2668    2686    2709    2726    2741
             2759    2776    2801    2826    2845    2869    2892    2908    2926    2952    2980    2998    3016    3034    3051
             3075    3091    3109    3128    3160    3197    3234    3271    3308    3333    3358    3383    3409    3435    3461
             3487    3513    3535    3558    3594    3630    3648    3666    3684    3707    3723    3751    3765    3779    3793
             3808    3823    3838    3852    3867    3882    3897    3912    3927    3942    3956    3971    3986    4001    4016
             4030    4045    4060    4075    4101    4119    4136    4172    4193    4217    4248    4270    4310    4331    4371
             4394    4416    4435    4459    4480    4508    4529    4547    4565    4645    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    5621    5649    5677    5705    5733    5763    5789    5824    5850
             5881    5907    5937    5962    5988    6015    6041    6066    6092    6118    6144    6170    6196    6222    6247
             6273    6299    6325    6351    6376    6402    6428    6454    6480    6506    6532    6558    6584    6612    6641
             6673    6705    6732    6759    6786    6813    6840    6867    6893    6920    6947    6974    7001    7028    7055
             7082    7109    7136    7163    7190    7217    7244    7270    7296    7322    7348    7374    7401    7423    7449
```

```
                7470    7497    7523    7549    7570    7597    7624    7648    7672    7697    7721    7746    7765    7785    7806
                7827    7856    7885    7914    7943    7972    8001    8030    8059    8081    8103    8125    8147    8173    8200
                8227    8254    8283    8312    8345    8374    8401    8428    8455    8482    8509    8538    8567    8595    8622
                8649    8676    8706    8737    8763    8789    8815    8841    8862    8883    8915    8947    8979    9011    9048
                9085    9122    9159    9196    9233    9274    9311    9343    9375    9407    9439    9460    9481    9502    9523
                9545    9566    9593    9621    9641    9660    9684    9708    9729    9756    9783    9810    9837    9859    9881
                9903    9925    9947    9969    9991   10018   10049   10076   10103   10131   10158   10185   10212   10238   10267
               10296   10325   10354   10383   10411   10440   10469   10497   10524   10551   10579   10606   10633   10659   10686
               10713   10740   10769   10798   10828   10858   10888   10922   10952   10982   11011   11030   11051   11073   11095
               11117   11139   11161   11183   11204   11226   11247   11269   11291   11313   11340   11362   11394   11420   11451
               11473   11503   11529   11555   11581   11611   11641   11669   11700   11729   11754   11783   11816   11848   11881
               11909   11943   11959   11986   12005   12024   12043   12062   12103   12140   12177   12237   12297   12321   12339
               12357   12387   12419   12452   12485   12542   12599   12630   12661   12692   12723   12754   12790   12822   12866
               12953   12973   13013   13051   13076   13101   13126   13150   13189   13227   13279   13339   13399   13429   13461
               13493   13516   13539   13561   13592   13629   13664   13699   13725   13752   13779   13809   13849   13879   13894
               13919   13935   13952   13972   14032   14092   14152   14212   14257   14301   14331   14362   14398   14434   14470
               14506   14537   14561   14590   14619   14648   14670   14716   14740   14765   14789   14817   14846   14891   14929
               14980   15026   15061   15148   15209   15279   15316   15392   15431   15553   15640
$$SET          16358#  16367   16368   16369
$$SKIP             1#    830#   1947    1961    1974    1987    2002    2016    2034    2050    2066    2088    2110    2128    2146
                2165    2184    2202    2224    2247    2264    2286    2312    2341    2361    2379    2395    2413    2432    2453
                2474    2494    2515    2537    2560    2583    2606    2644    2663    2681    2704    2721    2754    2771    2796
                2821    2840    2865    2887    2904    2921    2947    2975    2993    3011    3029    3046    3086    3104    3123
                3155    3191    3228    3265    3302    3329    3354    3379    3404    3430    3456    3482    3508    3531    3553
                3644    3662    3680    3702    3718    3746    3760    3774    3788    3803    3818    3833    3847    3862    3877
                3892    3907    3922    3937    3951    3966    3981    3996    4011    4025    4040    4055    4070    4096    4114
                4131    4167    4188    4212    4243    4264    4304    4365    4411    4431    4475    4521    4541    4559    4655
                4669    4689    4711    4731    4753    4767    4782    4797    4812    4827    4841    4856    4871    4886    4912
                4933    4959    4976    5028    5045    5071    5089    5115    5141    5193    5220    5246    5272    5298    5325
                5352    5379    5406    5432    5458    5484    5510    5536    5562    5589    5616    5644    5672    5700    5728
                5758    5784    5819    5845    5876    5902    5932    5957    5984    6011    6037    6062    6088    6114    6140
                6166    6192    6218    6243    6269    6295    6321    6347    6372    6398    6424    6450    6476    6502    6528
                6554    6580    6607    6636    6668    6700    6727    6754    6781    6808    6835    6862    6888    6915    6942
                6969    6996    7023    7050    7077    7104    7131    7158    7185    7212    7239    7266    7292    7318    7344
                7370    7396    7418    7444    7465    7492    7518    7544    7565    7593    7620    7644    7668    7693    7717
                7742    7761    7781    7801    7822    7852    7881    7910    7939    7968    7997    8026    8055    8076    8098
                8120    8142    8169    8196    8223    8250    8278    8307    8340    8369    8397    8424    8451    8478    8505
                8533    8562    8590    8618    8645    8672    8701    8732    8759    8785    8811    8837    8857    8878    8909
                8941    8973    9005    9042    9079    9116    9153    9190    9227    9268    9305    9337    9369    9401    9433
                9455    9476    9497    9518    9539    9561    9589    9617    9637    9656    9680    9704    9724    9752    9778
                9805    9832    9854    9876    9898    9920    9942    9964    9986   10014   10045   10072   10099   10126   10154
               10181   10208   10234   10262   10291   10320   10349   10378   10407   10435   10464   10492   10520   10547   10575
               10602   10629   10656   10682   10709   10736   10764   10793   10823   10853   10883   10917   10947   10977   11006
               11026   11046   11068   11090   11112   11134   11156   11178   11199   11221   11242   11264   11282   11287   11304
               11309   11326   11336   11353   11358   11375   11385   11388   11407   11412   11415   11433   11437   11447   11464
               11469   11486   11496   11499   11516   11521   11524   11542   11547   11550   11568   11573   11576   11594   11597
               11602   11605   11624   11627   11632   11635   11660   11773   11802   11838   11867   11900   11929   11954   11980
               11983   12001   12020   12039   12058   12095   12132   12169   12196   12204   12216   12225   12256   12264   12276
               12285   12334   12352   13184   13222   13261   13330   13390   13615   13650   13685   14248   14292   14327   14358
               14394   14430   14466   14502   14533   14557   14586   14615   14644   14666   14709   14737   14762   14786   14814
               14843   14875
.EQUAT             1#    698#    720
.HEADE             1#    698#    699
.KT11              1#
.SETUP             1#
.SWRHI             1#    698#    708
```

```
.SWRLO    719#
.$ACT1      1#    698#    864
.$APTB      1#    698#    930#
.$APTH      1#    698#    842
.$APTY      1#    698#  16277
.$ASTA      1#
.$CATC      1#    698#    830
.$CMTA      1#    698#    874
.$DB2D      1#
.$DB2O      1#
.$DIV       1#
.$EOP       1#    698#  15683
.$ERRO      1#    698#  16000
.$ERRT      1#    698#  16063
.$MULT      1#
.$POWE      1#    698#  15724
.$RAND      1#
.$RDDE      1#
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#    698#  15918
.$SIZE      1#
.$SUPR      1#
.$TRAP      1#    698#  16335
.$TYPB      1#
.$TYPD      1#    698#
.$TYPE      1#    698#  16119
.$TYPO      1#    698#  16199
.$40CA      1#
.1170       1#


. ABS.  070034     000


 ERRORS DETECTED:  0

 CQKDAC.BIN,CQKDAC.LST/CRF/SOL/NL:TOC=CQKDAC.SML,CQKDAC.P11
 RUN-TIME: 66 79 6 SECONDS
 RUN-TIME RATIO: 868/151=5.7
 CORE USED:  39K  (77 PAGES)
```