KD11-K

BASIC LOGIC TESTS

MD-11-DQKDA-A

EP-DQKDA-A-DL-A

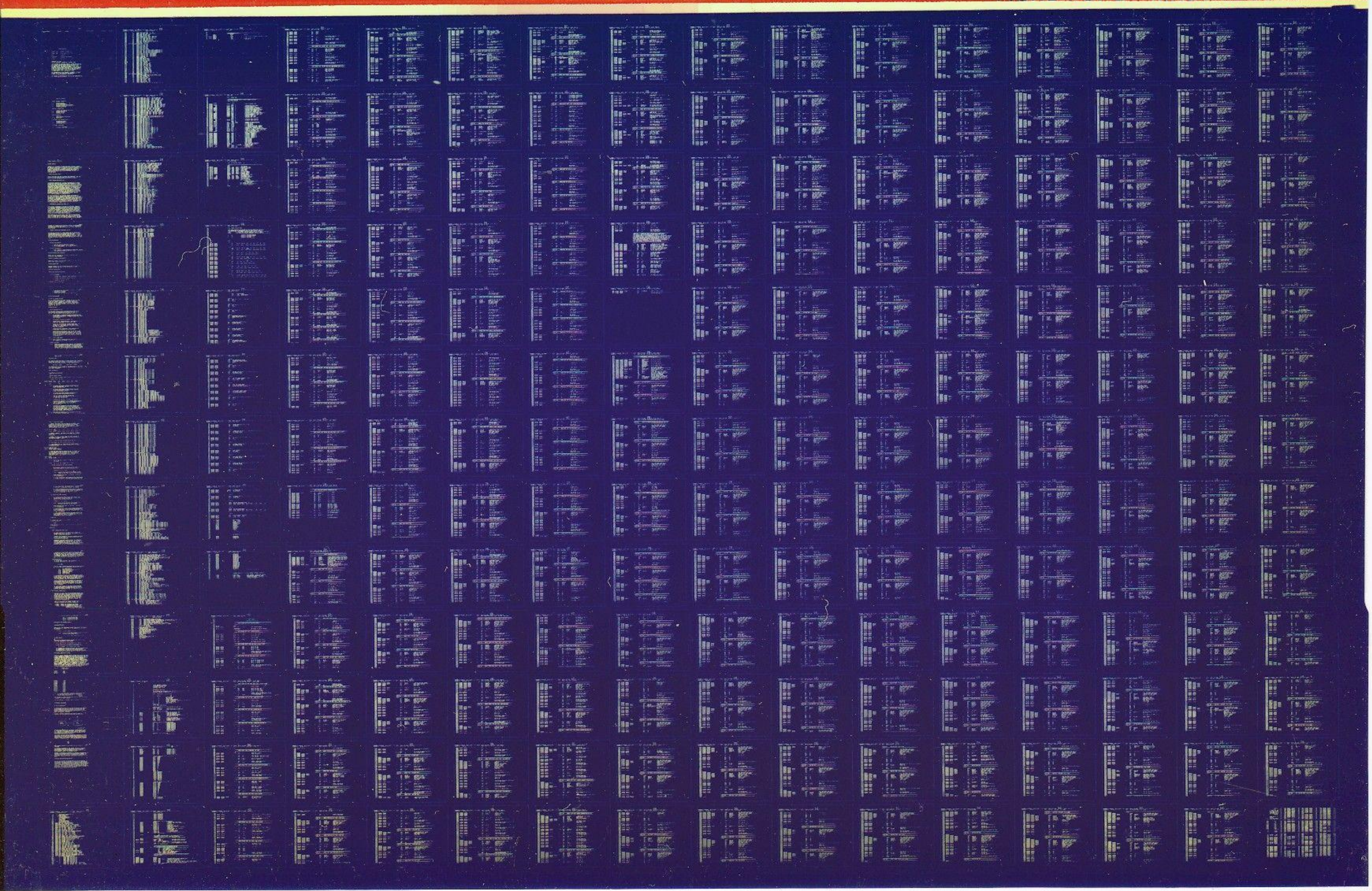
COPYRIGHT © 1977

FICHE 1 OF 2

APR 1977

digital

MADE IN USA



KD11-K

BASIC LOGIC TESTS

MD-11-DQKDA-A

EP-DQKDA-A-DL-A

COPYRIGHT © 1977

FICHE 2 OF 2

APR 1977

digital

MADE IN USA



EOF1DQFPORSBQ411 [illegible] [illegible]T LISTING PDP10 411 HDR1DQKDAASEQ 00010000 770323
.REM %

IDENTIFICATION

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

PRODUCT NAME: KD11-K BASIC LOGIC TESTS

DATE: 01-FEB-77

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: JACK RICH

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

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

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

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

TABLE OF CONTENTS

1.0 GENERAL PROGRAM INFORMATION

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

2.0 OPERATING INSTRUCTIONS

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

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES
3.2 ERROR HALTS

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS
4.2 PROGRESS REPORTS
4.3 MAINTENANCE BREAKPOINT FEATURE

5.0 MAINTENANCE PROCEDURES

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

1.0 GENERAL PROGRAM INFORMATION

1.1 PROGRAM PURPOSE

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

A. BASIC CPU TESTS (BCPT)

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

B. BASIC INSTRUCTION TESTS (BIT)

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

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

C. COMPREHENSIVE INSTRUCTION TESTS (CIT)

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

UPON DETECTION OF A LOGIC FAULT, EACH TEST IN THIS SECTION CALLS AN "ERROR SERVICE" ROUTINE THAT LOGS THE ERROR AND REPORTS IT AS HARD COPY ON THE CONSOLE TERMINAL DEVICE. THE ERROR SERVICE ROUTINE ALSO FACILITATES USER CONTROL OF THE PROGRAM

SEQUENCE VIA CONSOLE SWITCH REGISTER OPTIONS. AFTER REPORTING THE ERROR THE PROGRAM CONTINUES ON IN ITS NORMAL SEQUENCE UNLESS MODIFIED BY THE USER ACTIVATING THE "LOCK ON HARD ERROR" SWITCH OPTION.

D. COMBINED INSTRUCTION EXERCISER (IEX)

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

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

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

B. SOFTWARE REQUIREMENTS

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

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY REQUIREMENTS

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

1.5 FAILURE ASSUMPTIONS

"DQKDAA" 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 F01
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 "DQKDAA" WILL NOT PERMIT SIMULTANEOUS RESIDENCY OF THE UPDATE PROGRAMS. SUFFICIENT FREE CORE IS AVAILABLE FOR THE "XXDP" MONITOR SO THAT "DQKDAA" 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 [illegible] TS" THROUGHOUT THE PROGRAM. THIS ENABLES RAPIDLY "[illegible]MING 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 DQKDAA 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 P INTS IN THE COMPREHENSIVE INSTR-
UCTION TESTS THE LINE CLOCK VECTOR (100/102) AND THE DL11
VECTORS (60/62 - 64/66) ARE SET UP TO CHECK INTERRUPTS
FROM THESE DEVICES. ALL OTHER VECTORS REMAIN SET UP TO
"CATCH" UNEXPECTED TRAPS OR INTERRUPTS BY HALTING.

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

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

3. CATASTROPHIC ERROR HALTS

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

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

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

THE STACK PROVIDES THE FOLLOWING ADDITIONAL INFORMATION:

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

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

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

```
PASCNT = XXXXXX  ERRCNT = YYYYYY
```

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

YYYYYY IS THE TOTAL ERROR COUNT IN OCTAL

4.2 PROGRESS REPORTS

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

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

B. PROGRAM IDENTIFICATION MESSAGE AS DESCRIBED BELOW:

```
MD-11-DQKDAA   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	4302
BIT01=1	6266
BIT02=1	10606

```
BIT03=1          11736
BIT04=1          14316
BIT05=1          17056
BIT06=1          21502
BIT07=1          24310
BIT08=1          27122
BIT09=1          32116
BIT10=1          34602
BIT11=1          37412
BIT12=1          42064
BIT13=1          46102
BIT14=1          52542
BIT15=1          55366
```

NOTE: IF THE USER DEPOSITED A 000400(8) IN LOCATION "BPTLOC"
ONLY ONE HALT WOULD OCCUR AND AT THAT TIME THE
DISPLAY SHOULD CONTAIN 27114.

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

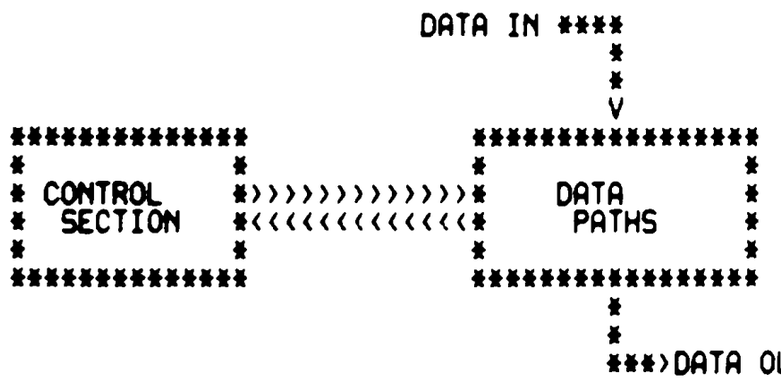
LOCATION "BPTLOC" IS PROGRAM LOADED AS 000000 TO INHIBIT ANY HALTS.

5.0 MAINTENANCE PROCEDURES

5.1 THE KD11-K PROCESSOR

THE PROCEDURES OUTLINED IN THIS SECTION ASSUME THAT "DQKDAA"
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:



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 POSSIBLE DEPENDENT UPON OF THE PROGRAM CODING.

"DQKDAA" 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 B03M2 AND MAY BE USED IN CONJUNTION WITH THE PROGRAM LOOPING FEATURES TO PROBE THE KDII-K DURING THE FAILING TEST.

%

```
   13          OPERATIONAL SWITCH SETTINGS
   25          BASIC DEFINITIONS
  135          TRAP CATCHER
  144          STARTING ADDRESS(ES)
  147          APT PARAMETER BLOCK
  169          ACT11 HOOKS
  179          COMMON TAGS
  235          APT MAILBOX-ETABLE
  262          ERROR POINTER TABLE
  576  BT001 "BR" TEST -POSITIVE OFFSET
  585  BT002 "BR" TEST - NEGATIVE OFFSET
  599  BT003 "BASIC COND. BR" TEST - FLAGS CLEARED
  612  BT004 "SCC AND COND. BR'S" TEST - FLAGS SET
  627  BT005 "CCC AND COND. BR'S" TEST - FLAGS CLEARED
  642  BT006 "CLR  %R" TEST - SETS THE "Z" BIT
  655  BT007 "TST  %R" TEST - USING THE CLR
  670  BT010 "COM  %R" TEST - SHOULD SET "N" AND "C"
  685  BT011 "COM  %R AND ADC  %R" TEST
  701  BT012 "MOV  #N,R" TEST WITH N=177777,[R]=000000
  716  BT013 "MOV  #N,R" TEST WITH N=000000,[R]=177777
  733  BT014 "CLR  (R)" TEST - [R] = 177776
  748  BT015 "CLR  (R)+" TEST - [R] = 177776
  769  BT016 "COM  (R)" TEST - [R] = 177776
  786  BT017 "COM  (R0)+" TEST - [R0] = 177776
  811  BT020 "MOV  RA,RB" TEST - WITH [RA]=177777,[RB]=000000
  833  BT021 "MOV  RA,RB" TEST WITH [RA]=000000,[RB]=177777
  856  BT022 "MOV  #N,@#A" TEST WITH N=17,A=177776
  872  BT023 "MOV  RA,(RB)+" TEST WITH [RA]=17,[RB]=177776
  896  BT024 "CMP  #N,@#A" TEST WITH N=(A)
  911  BT025 "CMP  #N,@#A" WITH N >  (A)
  924  BT026 "CMP  #N,@#A" WITH N <  (A)
  938  BT027 "CMP  R,#N" TEST WITH [R]=N
  952  BT030 "CMP  R,#N" TEST WITH [R] > N
  965  BT031 "CMP  R,#N" TEST WITH [R] < N
  979  BT032 "CMP  (RA)+,RB" TEST WITH [SOURCE]=[RB]
 1001  BT033 "CMP  (RA)+,RB" TEST WITH [SOURCE]>[RB]
 1022  BT034 "CMP  (RA)+,RB" TEST WITH [SOURCE]<[RB]
 1043  BT035 "CMP  RA,RB" TEST WITH [RA] = [RB]
 1057  BT036 "CMP  RA,RB" TEST WITH [RA] < [RB]
 1071  BT037 "CMP  RA,RB" TEST WITH [RA] > [RB]
 1086  BT040 "MOV  (RA),RB" TEST WITH [SOURCE]=[RB]=17
 1102  BT041 "MOV  (RA)+,RB" TEST WITH [SOURCE]=[RB]=17
 1125  BT042 "XOR  RA,RB" TEST WITH [RA] = [RB] = 000000
 1141  BT043 "XOR  RA,RB" TEST WITH [RA] = [RB] = 177777
 1158  BT044 "XOR  RA,RB" TEST WITH [RB]=052525,[RA]=125252
 1173  BT045 "XOR  RA,RB" TEST WITH [RA]=052525,[RB]=125252
 1189  BT046 GPR ADDRESS INTERRACTION TEST
 1237          T0      BASIC "BNE" TEST WITH Z=0
 1251          T1      BASIC "BNE" TEST WITH Z=1
 1266          T2      BASIC "BEQ" TEST WITH Z=1
 1279          T3      BASIC "BEQ" TEST WITH Z=0
 1294          T4      BASIC "BPL" TEST WITH N=1
 1310          T5      BASIC "BPL" TEST WITH N=0
 1325          T6      BASIC "MOV  (RA),RB" TEST - (RA)=177776
 1343          T7      BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
 1360          T10     BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
```

1378 T11 BASIC "CMP #N,R" TEST - N = [R]
1401 T12 BASIC "CMP #N,R" TEST - N NOT EQUAL TO [R]
1424 T13 BASIC "MOV RA,(RB)" TEST
1443 T14 BASIC "MOV #N,(R)" TEST
1462 T15 BASIC "MOVB #N,X(R)" TEST - DEST EVEN
1482 T16 BASIC "MOVB #N,X(R)" TEST - DEST ODD
1502 T17 BASIC "TST @#A" TEST WITH [A] GT 0
1520 T20 BASIC "TST @#A" TEST WITH [A] LT 0
1544 T21 BASIC "TST @#A" WITH [A] = 0
1568 T22 BASIC "BIT #N,@#A" WITH BIT SET IN "A"
1586 T23 BASIC "BIT #N,@#A" WITH BIT CLEAR IN "A"
1609 T24 BASIC "TST (R)+" TEST
1615 USER CONTROLLED BREAKPOINT -- BIT0
1636 T25 BASIC "TST -(R)" TEST
1667 T26 BASIC "COM @#A" TEST
1687 T27 BASIC "INC @#A" TEST
1706 T30 BASIC "DEC RN" TEST
1723 T31 BASIC "DEC @#A" TEST
1742 T32 BASIC "CLR X(R)" TESTS
1762 T33 BASIC "ASL RN" TEST WITH [DEST]=125252 AND C(0)
1784 T34 BASIC "ASL RN" TEST WITH [DEST]=052525 AND C(1)
1805 T35 BASIC "ROL RN" TEST WITH [DEST]=125252 AND C(0)
1827 T36 BASIC "ROL RN" TEST WITH [DEST]=052524 AND C(1)
1849 T37 BASIS "TSTB (R)" TEST - EVEN ADDRESS
1872 T40 BASIS "TSTB (R)" TEST - ODD ADDRESS
1897 T41 BASIC "TSTB @#A" TEST - EVEN ADDRESS
1921 T42 BASIC "TSTB @#A" TEST - ODD ADDRESS
1945 T43 BASIC "DECB 1(SP)"
1967 T44 BASIC "MOV @#A,R" TEST
1984 T45 BASIC "MOV #N,X(R)" TEST
2004 T46 BASIC "MOV #N,(R)" TEST
2023 T47 BASIC "MOV (RA)+,RB" TEST
2047 T50 BASIC "MOV @#A,@#B"
2065 T51 BASIC "MOV   X(R),PC" TEST
2081 T52 BASIC "MOV @#A,(R)" TEST
2100 T53 BASIC "MOV X(RA),RB" TEST
2118 T54 BASIC "MOV RA,-(RB)" TEST
2144 T55 BASIC "MOV   @#A,-(R)" TEST
2170 T56 BASIC "MOV (R),@#A" TEST
2190 T57 BASIC "MOV -(R),@#A" TEST
2215 T60 BASIC "MOV   (RA),RB" TEST
2239 T61 BASIC "MOV   X(RA),RB" TEST
2256 T62 BASIC "MOV   @X(RA),RB" TEST
2275 T63 BASIC "MOV (R)+,X(R)" TEST
2302 T64 BASIC "CMP R,@#A" TEST WITH [R] = [A]
2308 USER CONTROLLED BREAKPOINT -- BIT1
2331 T65 BASIC "CMP R,@#A" WITH [R] NOT EQUAL TO [A]
2350 T66 BASIC "BIS #N,@#A" TEST - N=177777,[A]=000000
2369 T67 BASIC "BIC #N,@#A" TEST
2388 T70 BASIC "BIC #N,R" TEST
2406 T71 BASIC "BIC #N,2(SP)" TEST
2431 T72 BASIC "ADD #N,RN " TEST
2448 T73 BASIC "ADD #N,(R) " TEST
2467 T74 BASIC "ADD #N,X(R)" TEST
2487 T75 BASIC "CMPB #N,(SP)+" TEST
2520 T76 BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST EVEN

2558 T77 BASIC "CMPB (R0)+,(R0)+" - SRC AND DEST ODD
2596 T100 BASIC "CMPB (R0)+,(R0)+" - SRC / EVEN,DEST / ODD
2634 T101 BASIC "CMPB (R0)+,(R0)+" - SRC / ODD,DEST / EVEN
2672 T102 BASIC "MOVB (R0)+,X(R0) - SRC EVEN / DEST EVEN
2698 T103 BASIC "MOVB (R0)+,X(R0) - SRC ODD / DEST ODD
2724 T104 BASIC "MOVB (R0)+,X(R0) - SRC EVEN / DEST ODD
2750 T105 BASIC "MOVB (R0)+,X(R0) - SRC ODD / DEST EVEN
2777 T106 BASIC "MOVB 2(R0),(R0)+" TEST - SRC EVEN / DEST EVEN
2804 T107 BASIC "MOVB 2(R0),(R0)+" TEST - SRC ODD / DEST EVEN
2831 T110 BASIC "MOVB 2(R0),(R0)+" TEST - SRC EVEN / DEST ODD
2858 T111 BASIC "MOVB 2(R0),(R0)+" TEST - SRC ODD / DEST ODD
2885 T112 BASIC "MOVB -(R0),R0" TEST - SRC EVEN ADDR
2908 T113 BASIC "MOVB -(R0),R0" TEST - SRC ODD ADDR
2932 T114 BASIC "MOVB (R0)+,-(SP)" TEST - SRC ADDR EVEN
2969 T115 BASIC "MOVB (R0)+,-(SP)" TEST - SRC ADDR ODD
3006 T116 BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST EVEN
3025 T117 BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST EVEN
3044 T120 BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST ODD
3063 T121 BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST ODD
3069 USER CONTROLLED BREAKPOINT -- BIT2
3087 T122 BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=0
3104 T123 BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLAG=1
3133 T124 BASIC BVC TEST WITH V=1
3148 T125 BASIC BVC TEST WITH V=0
3163 T126 BASIC BGE TEST WITH N,V = 00
3178 T127 BASIC BGE TEST WITH N,V = 01
3194 T130 BASIC BGE TEST WITH N,V = 10
3210 T131 BASIC BGE TEST WITH N,V = 11
3226 T132 BASIC BLT TEST WITH N,V = 00
3241 T133 BASIC BLT TEST WITH N,V = 01
3257 T134 BASIC BLT TEST WITH N,V = 10
3273 T135 BASIC BLT TEST WITH N,V = 11
3289 T136 BASIC BGT TEST WITH Z = 1 AND N,V = 01
3305 T137 BASIC BGT TEST WITH Z = 0 AND N,V = 01
3321 T140 BASIC BGT TEST WITH Z = 1 AND N,V = 00
3337 T141 BASIC BGT TEST WITH Z = 0 AND N,V = 00
3352 T142 BASIC BGT TEST WITH Z = 1 AND N,V = 01
3368 T143 BASIC BGT TEST WITH Z = 1 AND N,V = 10
3384 T144 BASIC BGT TEST WITH Z = 1 AND N,V = 11
3400 T145 BASIC BGT TEST WITH Z=0 AND N,V=11
3416 T146 BASIC BHI TEST WITH Z,C = 00
3431 T147 BASIC BHI TEST WITH Z,C = 01
3447 T150 BASIC BHI TEST WITH Z,C = 10
3463 T151 BASIC BHI TEST WITH Z,C = 11
3479 T152 BASIC NEG MODE 0 TEST : [DEST] GT 0
3506 T153 BASIC "SUB @#,@#" TEST
3525 T154 BASIC "SUB @#,RN" TEST
3543 T155 BASIC "RTS RN" TEST - <N:C> = 1111
3580 T156 BASIC "RTS PC" TEST
3602 T157 BASIC "JSR PC,@#A" TEST
3608 USER CONTROLLED BREAKPOINT -- BIT3
3627 T160 BASIC "RTI" TEST - N:C=0000
3659 T161 BASIC "RTI" TEST WITH N:C=1111
3682 T162 BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
3723 T163 BASIC "IOT" TEST - VERIFY LINKAGE TO SCOPE SERVICE
3745 T164 BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357

3786 T165 BASIC IOT TEST - VERIFY LOADING PSW WITH 000
3810 T166 BASIC "TRAP" TEST - LINKAGE TO PRINT ROUTINE
3831 T167 BASIC "EMT" TEST - LINKAGE TO ERROR SERVICE
3851 T170 BASIC TEST OF RSVD INSTR. TRAP LINKAGE
3876 T171 BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
3898 T172 BASIC TEST FOR ACCESSING DL11 REGISTERS
3927 T173 BASIC TEST OF DL11 - XCSR - READY(1)
3949 T174 BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
3968 T175 BASIC TEST OF DL11 XCSR - MAINT BIT = 1
3987 T176 BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
4068 T177 BCC TEST WITH C=1
4083 T200 BCC TEST WITH C=0
4098 T201 VERIFY NO BRANCH MICROROUTINE DOES NOT CLR FLAGS
4119 T202 VERIFY BRANCH MICROROUTINE DOES NOT CLR FLAGS
4142 T203 VERIFY NO BRANCH MICROROUTINE DOES NOT SET FLAGS
4163 T204 VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
4186 T205 BLE TEST WITH Z = 0, AND N,V = 00
4201 T206 BLE TEST WITH Z = 1 AND N,V = 00
4217 T207 BLE TEST WITH Z = 0 AND N,V = 01
4233 T210 BLE TEST WITH Z = 0 AND N,V = 10
4249 T211 BLE TEST WITH Z = 0 AND N,V = 11
4265 T212 BLOS TEST WITH Z,C = 00
4280 T213 BLOS TEST WITH Z,C = 01
4296 T214 BLOS TEST WITH Z,C = 10
4312 T215 BLOS TEST WITH Z,C = 11
4328 T216 SXT MODE 0 TEST WITH N = 0 AND C = 1
4355 T217 SXT MODE 0 TEST WITH N = 0 AND C = 0
4363 USER CONTROLLED BREAKPOINT -- BIT4
4377 T220 SXT MODE 0 TEST WITH N = 1 AND C = 1
4404 T221 SXT MODE 0 TEST WITH N = 1 AND C = 0
4422 T222 SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
4475 T223 SXT MODE 1 TEST WITH N = 0 AND C = 0
4493 T224 SXT MODE 1 TEST WITH N = 1 AND C = 1
4520 T225 SXT MODE 1 TEST WITH N = 1 AND C = 0
4539 T226 SWAB MODE 0 TEST WITH POS. RESULT
4566 T227 SWAB MODE 0 TEST WITH NEG. RESULT
4593 T230 SWAB MODE 1 AND 2 TEST WITH POS. RESULT
4646 T231 SWAB MODE 1 TEST WITH NEG. RESULT
4674 T232 NEG MODE 0 TEST : [DEST] = 0
4701 T233 NEG MODE 0 TEST : [DEST] LT 0
4728 T234 NEG MODE 0 TEST : [DEST] = 100000 (8)
4755 T235 NEG MODE 1 TEST : [DEST] = 0
4783 T236 NEG MODE 1 TEST : [DEST] GT 0
4811 T237 NEG MODE 1 TEST : [DEST] LT 0
4839 T240 NEG MODE 1 TEST: [DEST] = 100000 (8)
4867 T241 ROR TEST - DM0 - N:C = 1110
4894 T242 ROR TEST - DM0 - N:C = 1000
4921 T243 ROR TEST - DM0 - N:C = 0111
4948 T244 ASR TEST - DM0 - N:C = 1000
4975 T245 ASR TEST - DM0 - N:C = 0101
5002 T246 ASR TEST - DM0 - N:C = 1100
5029 T247 ROR TEST - DM1 - N:C = 1110
5057 T250 ROR TEST - DM1 - N:C = 1000
5086 T251 ROR TEST - DM1 - N:C = 0111
5115 T252 ASR TEST - DM1 - N:C = 1000
5144 T253 ASR TEST - DM1 - N:C = 1100

5173 T254 ASR TEST - DM1 - N:C = 0101
5202 T255 RORB TEST - DM2 - EVEN ADDRESS
5233 T256 RORB TEST - DM1 - EVEN ADDRESS
5260 T257 RORB TEST - DM2 - ODD ADDRESS
5288 USER CONTROLLED BREAKPOINT -- BIT5
5296 T260 RORB TEST - DM1 - ODD ADDRESS
5323 T261 ASRB TEST - DM2 - ODD ADDRESS
5355 T262 ASRB TEST - DM1 - ODD ADDRESS
5382 T263 ASRB TEST - DM2 - EVEN ADDRESS
5413 T264 ASRB TEST - DM1 - EVEN ADDRESS
5439 T265 TST DM0 TEST - N:C = 1011
5466 T266 TST DM0 TEST - N:C = 0100
5494 T267 CLR DM0 TEST - N:C = 1011
5521 T270 CLR DM0 TEST - N:C = 0000
5547 T271 COM DM0 TEST - N:C = 0110
5574 T272 COM DM0 TEST - N:C = 1001
5601 T273 INC DM0 TEST - N:C = 1011
5628 T274 INC DM0 TEST - N:C = 0100
5655 T275 DEC DM0 TEST - N:C = 1011
5682 T276 DEC DM0 TEST - N:C = 1100
5709 T277 DEC DM0 TEST - N:C = 0000
5735 T300 ASL DM0 TEST - N:C = 1000
5762 T301 ASL DM0 TEST - N:C = 0101
5789 T302 ASL DM0 TEST - N:C = 0010
5816 T303 ROL DM0 TEST - N:C = 1101
5843 T304 ROL DM0 TEST - N:C = 0101
5869 T305 ROL DM0 TEST - N:C = 0010
5896 T306 ADC DM0 TEST - N:C = 0101
5923 T307 ADC DM0 TEST - N:C = 1011
5950 T310 ADC DM0 TEST - N:C = 1010
5977 T311 SBC DM0 TEST - N:C = 1011
6004 T312 SBC DM0 TEST - N:C = 0101
6031 T313 SBC DM0 TEST - N:C = 1110
6058 T314 SBC DM0 TEST - N:C = 0111
6085 T315 TST DM1 TEST - N:C = 1011
6114 T316 TST DM1 TEST - N:C = 0100
6144 T317 CLR DM1 TEST - N:C = 1011
6152 USER CONTROLLED BREAKPOINT -- BIT6
6177 T320 CLR DM2 TEST - N:C = 0000
6210 T321 COM DM1 TEST - N:C = 0110
6238 T322 COM DM1 TEST - N:C = 1001
6266 T323 INC DM1 TEST - N:C = 1011
6294 T324 INC DM1 TEST - N:C = 0100
6322 T325 DEC DM1 TEST - N:C = 1011
6350 T326 DEC DM1 TEST - N:C = 1100
6378 T327 DEC DM1 TEST - N:C = 0000
6405 T330 ASL DM1 TEST - N:C = 1000
6433 T331 ASL DM1 TEST - N:C = 0101
6461 T332 ASL DM1 TEST - N:C = 0010
6489 T333 ROL DM1 TEST - N:C = 1101
6517 T334 ROL DM1 TEST - N:C = 0101
6545 T335 ROL DM1 TEST - N:C = 0010
6573 T336 ADC DM1 TEST - N:C = 0101
6601 T337 ADC DM1 TEST - N:C = 1011
6629 T340 ADC DM1 TEST - N:C = 1010
6657 T341 SBC DM1 TEST - N:C = 1011

6685 T342 SBC DM1 TEST - N:C = 0101
6713 T343 SBC DM1 TEST - N:C = 1110
6741 T344 SBC DM1 TEST - N:C = 0111
6769 T345 NEGB - MODE 0 TEST - N:C = 0110
6796 T346 NEGB - MODE 0 TEST - N:C = 0011
6823 T347 NEGB - MODE 0 TEST - N:C = 1101
6850 T350 CLRB - MODE 0 TEST - N:C = 1011
6877 T351 CLRB - MODE 0 TEST - N:C = 0100
6904 T352 CLRB TEST - DM2 - 000 ADDRESS
6932 T353 CLRB TEST - DM1 - 000 ADDRESS
6955 T354 CLRB TEST - DM2 - EVEN ADDRESS
6982 T355 CLRB TEST - DM1 - EVEN ADDRESS
7004 T356 NEGB TEST - DM2 - 000 ADDRESS
7032 T357 NEGB TEST - DM1 - 000 ADDRESS
7040 USER CONTROLLED BREAKPOINT -- BIT7
7059 T360 NEGB TEST - DM2 - EVEN ADDRESS
7086 T361 NEGB TEST - DM1 - EVEN ADDRESS
7108 T362 ADD TEST - SM0,DM0 - N:C = 1010
7136 T363 ADD TEST - SM0,DM0 - N:C = 0101
7164 T364 ADD SM1,DM0 TEST
7189 T365 ADD SM2,DM0 TEST
7214 T366 ADD SM3,DM0 TEST
7240 T367 ADD SM4,DM0 TEST
7265 T370 ADD SM5,DM0 TEST
7291 T371 ADD SM6,DM0 TEST
7311 T372 ADD SM7,DM0 TEST
7332 T373 ADD SM1,DM1 TEST
7354 T374 ADD SM2,DM1 TEST
7376 T375 ADD SM1,DM2 TEST
7406 T376 ADD SM2,DM2 TEST
7436 T377 ADD SM1,DM3 TEST
7466 T400 ADD SM2,DM3 TEST
7496 T401 ADD SM1,DM4 TEST
7526 T402 ADD SM2,DM4 TEST
7556 T403 ADD SM1,DM5 TEST
7586 T404 ADD SM2,DM5 TEST
7616 T405 ADD SM1,DM6 TEST
7639 T406 ADD SM2,DM6 TEST
7662 T407 ADD SM1,DM7 TEST
7685 T410 ADD SM2,DM7 TEST
7708 T411 "XOR RA,RB" TEST - A=B=000000 N:C=1010
7735 T412 "XOR RA,RB" TEST - A=B=177777 N:C=0101
7763 T413 "XOR RA,RB" TEST - A=125252,B=052525 N:C=0110
7791 T414 "XOR RA,RB" TEST - A=052525,B=125252 N:C=1001
7819 T415 "XOR RA,(RB)" TEST - A=B=000000 N:C=1010
7849 T416 "XOR RA,(RB)" TEST - A=B=177777 N:C=0101
7879 T417 "XOR RA,(RB)" TEST - A=125252,B=052525 N:C=0110
7887 USER CONTROLLED BREAKPOINT -- BIT8
7913 T420 "XOR RA,(RB)" TEST - A=052525,B=125252 N:C=1001
7943 T421 SUB TEST SM0,DM0 - (SRC) = (DEST) = +,+
7971 T422 SUB TEST SM0,DM0 - (SRC) = (DEST) = -,-
7999 T423 SUB TEST SM0,DM0 - (SRC) = (DEST) = -,+
8027 T424 SUB TEST SM0,DM0 (SRC) = -(DEST) = +,-
8055 T425 SUB TEST SM0,DM0 - "V" BIT SETS
8083 T426 SUB TEST - SM0,DM1 - N:C = 0110
8113 T427 SUB TEST - SM0,DM1 - N:C = 1010

8143 T430 SUB TEST - SM0,DM1 - N:C = 0000
8172 T431 SUB TEST - SM1,DM0 - N:C = 0110
8200 T432 SUB TEST - SM1,DM0 - N:C = 1010
8228 T433 SUB TEST - SM1,DM0 - N:C = 0000
8256 T434 SUB SM1,DM1 TEST - N:C = 0110
8287 T435 SUB SM1,DM2 TEST - N:C = 0110
8319 T436 NEG DM2 TEST
8346 T437 NEG DM3 TEST
8373 T440 NEG DM4 TEST
8400 T441 NEG DM5 TEST
8427 T442 NEG DM6 TEST
8449 T443 NEG DM7 TEST
8471 T444 MOV SM1,DM1 TEST - N:C = 0100
8504 T445 MOV SM2,DM1 TEST - N:C = 0100
8537 T446 MOV SM1,DM1 TEST - N:C = 1011
8570 T447 MOV SM2,DM1 TEST - N:C = 1011
8603 T450 MOV SM1,DM2 TEST - N:C = 0100
8641 T451 MOV SM2,DM2 TEST - N:C = 0100
8679 T452 MOV SM1,DM3 TEST - N:C = 0100
8717 T453 MOV SM2,DM3 TEST - N:C = 0100
8755 T454 MOV SM1,DM4 TEST - N:C = 0100
8793 T455 MOV SM2,DM4 TEST - N:C = 0100
8831 T456 MOV SM1,DM5 TEST - N:C = 0100
8839 USER CONTROLLED BREAKPOINT -- BIT9
8873 T457 MOV SM2,DM5 TEST - N:C = 0100
8911 T460 MOV SM1,DM6 TEST - N:C = 0100
8944 T461 MOV SM2,DM6 TEST - N:C = 0100
8977 T462 MOV SM1,DM7 TEST - N:C = 0100
9010 T463 MOV SM2,DM7 TEST - N:C = 0100
9043 T464 MOV SM0,DM1 TEST
9065 T465 MOV SM0,DM2 TEST
9087 T466 MOV SM0,DM3 TEST
9109 T467 MOV SM0,DM4 TEST
9131 T470 MOV SM0,DM5 TEST
9154 T471 MOV SM0,DM6 TEST
9176 T472 MOVB TEST - SM0,DM0 - EXTEND 1'S
9204 T473 MOVB TEST - SM0,DM0 - EXTEND 0'S
9233 T474 MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
9254 T475 MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
9274 T476 MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
9299 T477 MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
9324 T500 MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
9346 T501 MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
9374 T502 MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
9402 T503 MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
9430 T504 MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
9458 T505 MOVB TEST - SM1,DM6 - SRC ADR ODD / DST ADR EVEN
9481 T506 MOVB TEST - SM1,DM7 - SRC ADR ODD / DST ADR EVEN
9504 T507 MOVB SM0,DM1 TEST
9527 T510 MOVB SM0,DM2 TEST
9550 T511 MOVB SM0,DM3 TEST
9573 T512 MOVB SM0,DM4 TEST
9596 T513 MOVB SM0,DM6 TEST
9619 T514 BIS TEST - SM0,DM0 - N:C = 0111
9647 T515 BIS TEST - SM0,DM0 - N:C = 1000
9655 USER CONTROLLED BREAKPOINT -- BIT10

```
  9679          T516    BIC TEST - SM0,DM0 - N:C = 0111
  9707          T517    BIC TEST - SM0,DM0 - N:C = 1000
  9735          T520    BIT TEST - SM0,DM0 - N:C = 0111
  9764          T521    BIT TEST - SM0,DM0 - N:C = 1000
  9792          T522    CMP TEST - SM0,DM0 - N:C = 0110
  9820          T523    CMP TEST - SM0,DM0 - N:C = 1010
  9848          T524    CMP TEST - SM0,DM0 - N:C = 0000
  9875          T525    BIS TEST - SM0,DM1 - N:C = 0111
  9905          T526    BIS TEST - SM0,DM1 - N:C = 1000
  9935          T527    BIC TEST - SM0,DM1 - N:C = 0111
  9965          T530    BIC TEST - SM0,DM1 - N:C = 1000
  9995          T531    BIT TEST - SM0,DM1 - N:C = 0111
 10025          T532    BIT TEST - SM0,DM1 - N:C = 1000
 10054          T533    CMP TEST - SM0,DM1 - N:C = 1010
 10084          T534    CMP TEST - SM0,DM1 - N:C = 0110
 10114          T535    CMP TEST - SM0,DM1 - N:C = 0000
 10143          T536    BIS TEST - SM1,DM0 - N:C = 0111
 10171          T537    BIS TEST - SM1,DM0 - N:C = 1000
 10199          T540    BIC TEST - SM1,DM0 - N:C = 0111
 10228          T541    BIC TEST - SM1,DM0 - N:C = 1000
 10256          T542    BIT TEST - SM1,DM0 - N:C = 0111
 10284          T543    BIT TEST - SM1,DM0 - N:C = 1000
 10311          T544    CMP TEST - SM1,DM0 - N:C = 0110
 10339          T545    CMP TEST - SM1,DM0 - N:C = 1010
 10367          T546    CMP TEST - SM1,DM0 - N:C = 0000
 10395          T547    BIS  SM1,DM1 TEST - N:C = 0111
 10425          T550    BIS  SM1,DM1 TEST - N:C = 1000
 10455          T551    BIC  SM1,DM1 TEST - N:C = 0111
 10486          T552    BIC  SM1,DM1 TEST - N:C = 1000
 10517          T553    BIT  SM1,DM1 TEST - N:C = 1000
 10548          T554    BIT  SM1,DM1 TEST - N:C = 0111
 10556    USER CONTROLLED BREAKPOINT -- BIT11
 10583          T555    CMP  SM1,DM1 TEST - N:C = 1010
 10614          T556    CMP  SM1,DM1 TEST - N:C = 0110
 10645          T557    CMP  SM1,DM1 TEST - N:C = 0000
 10675          T560    BISB  SM1,DM0 TEST - SOURCE ADDR ODD
 10695          T561    BISB  SM1,DM1 TEST - SOURCE ADDR ODD
 10717          T562    BISB  SM1,DM2 TEST - SOURCE ADDR ODD
 10740          T563    BISB  SM1,DM3 TEST - SOURCE ADDR ODD
 10763          T564    BISB  SM1,DM4 TEST - SOURCE ADDR ODD
 10786          T565    BISB  SM1,DM5 TEST - SOURCE ADDR ODD
 10809          T566    BISB  SM1,DM6 TEST - SOURCE ADDR ODD
 10832          T567    BISB  SM1,DM7 TEST - SOURCE ADDR ODD
 10855          T570    BISB  SM0,DM2 TEST - DEST ADDR EVEN
 10877          T571    BISB  SM0,DM1 TEST - DEST ADDR ODD
 10900          T572    BISB  SM0,DM1 TEST - DEST ADDR EVEN
 10922          T573    BISB  SM1,DM1 TEST - DEST ADDR ODD
 10945          T574    JMP MODE 1 TEST, FLAGS = 1111
 10968          T575    JMP MODE 1 TEST, FLAGS = 0000
 10991          T576    JMP MODE 2 TEST; FLAGS = 1111
 11019          T577    JMP MODE 2 TEST; FLAGS = 0000
 11042          T600    JMP TEST MODE 3; FLAGS = 1111
 11075          T601    JMP TEST MODE 3; FLAGS = 0000
 11102          T602    JMP TEST MODE 4; FLAGS = 1111
 11134          T603    JMP TEST MODE 4; FLAGS = 0000
 11157          T604    JMP TEST MODE 5; FLAGS = 1111
```

11188 T605 JMP TEST MODE 5; FLAG = 0000
11215 T606 JMP TEST MODE 6; FLAGS = 1111
11242 T607 JMP TEST MODE 6; FLAGS = 0000
11269 T610 JMP TEST MODE 7; FLAGS = 1111
11300 T611 JMP TEST MODE 7; FLAGS = 0000
11331 T612 JSR MODE 1 TEST - LOAD PC / PUSH SP
11360 T613 JSR MODE 1 TEST - CHECK RN AND OLD PC
11392 T614 JSR MODE 1 TEST - N:C = 0000
11400 USER CONTROLLED BREAKPOINT -- BIT12
11422 T615 JSR MODE 1 TEST - N:C = 1111
11448 T616 JSR MODE 2 TEST
11478 T617 JSR MODE 3 TEST
11512 T620 JSR MODE 4 TEST
11545 T621 JSR MODE 5 TEST
11579 T622 JSR MODE 6 TEST
11608 T623 JSR MODE 7 TEST
11643 T624 SOB TEST, [R] = 1, NO BRANCH
11660 T625 SOB TEST, [R] = 5, BRANCH 4 TIMES
11688 T626 SOB TEST, [R] = 1, FLAGS = 1111
11708 T627 SOB TEST, [R] = 1, FLAGS = 0000
11728 T630 SOB TEST, [R] = 5, FLAGS = 1111
11748 T631 SOB TEST, [R] = 5, FLAGS = 0000
11768 T632 RTS TEST - N:C = 0000
11810 T633 RTT TEST - N:C = 1111
11848 T634 RTT TEST - N:C = 0000
11886 T635 MARK INSTRUCTION TEST - N:C=0000
11947 T636 MARK INSTRUCTION TEST - N:C=1111
12008 T637 BASIC LINE CLOCK RESPONSE TEST
12033 T640 LINE CLOCK TEST - LKCSR BIT 7 SET
12052 T641 LINE CLOCK TEST - LKCSR BIT 6 CLEAR
12071 T642 LINE CLOCK TEST - LKCSR BIT 6 SET
12102 T643 LINE CLK BASIC INTERRUPT TEST
12135 T644 RESET TEST - N:C = 1111
12169 T645 RESET TEST - N:C = 0000
12203 T646 WAIT INSTRUCTION TEST - [PSW] = 151
12261 T647 WAIT INSTRUCTION TEST - [PSW] = 010
12319 T650 BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
12351 T651 BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
12383 T652 BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
12415 T653 BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
12447 T654 BR PRIORITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12479 T655 BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12487 USER CONTROLLED BREAKPOINT -- BIT13
12516 T656 BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12549 T657 "CLR @#PSW" ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12594 T660 "BR6 VS BR4" PRIORITY ARBITRATION TEST
12682 T661 "BPT" TRAP LINKAGE TEST
12703 T662 RED ZONE OVERFLOW TEST - MOV R,-(SP)
12744 T663 YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
12783 T664 YELLOW ZONE OVERFLOW TEST - (CMP R0,-(SP))
12809 T665 YELLOW ZONE OVERFLOW TEST - (BIT R0,-(SP))
12835 T666 YELLOW ZONE OVERFLOW TEST - (TST -(SP))
12861 T667 ODD ADDRESS ERROR TEST - SUB RA,(RB) - (RB) = ODD
12886 T670 TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
12926 T671 TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
12965 T672 TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES

13018 T673 TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
13079 T674 TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
13140 T675 TEST THAT JSR CAN CAUSE OVERFLOW TRAP
13171 T676 TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
13204 T677 TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
13237 T700 ILLEGAL INSTRUCTION TEST - JSR RN,%R
13261 T701 ILLEGAL INSTRUCTION TEST - JMP %R
13285 T702 BUS TIMEOUT TRAP TEST - TST (R)
13308 T703 "T" BIT TRAP TEST
13340 T704 TEST PUSH INTO PSW WITH (SP) = 000000
13378 T705 TEST PUSH INTO SR WITH (SP) = 177572
13414 T706 TEST PUSH INTO SLR WITH (SP) = 177776
13450 T707 RSVD INSTRUCTION TEST - 000007 THRU 000077
13477 T710 RSVD INSTRUCTION TEST - 000210 THRU 000237
13505 T711 RSVD INSTRUCTION TEST - 007000 THRU 007777
13533 T712 RSVD INSTRUCTION TEST - 075000 THRU 076777
13564 T713 RSVD INSTRUCTION TEST - 106400 THRU 107777
13605 T714 BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
13636 T715 BUT SERVICE TEST - (RTI)
13652 T716 BUT SERVICE TEST - (JSR %R,@A)
13660 USER CONTROLLED BREAKPOINT -- BIT14
13678 T717 BUT SERVICE TEST - (JMP A)
13695 T720 BUT SERVICE TEST - (JMP @A)
13713 T721 BUT SERVICE TEST - (RTS PC)
13734 T722 ALU ADD FUNCTION TEST
13795 T723 ALU SUB FUNCTION TEST
13856 T724 ALU "AND" FUNCTION TEST USING BIC INSTRUCTION
13917 T725 ALU "OR" FUNCTION TEST USING BIS INSTRUCTION
13978 T726 INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
14024 T727 INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
14069 T730 MUL RA,RB TEST ; N:C = 1111
14100 T731 MUL (RA),RB TEST ; N:C = 0000-SET C
14132 T732 MUL (RA)+,RB TEST ; N:C = 0000-SET Z
14169 T733 MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST = -,+
14206 T734 MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK = +,-
14243 T735 MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
14280 T736 MUL X(RA),RB TEST ; N:C = 1111 TO 0100
14312 T737 MUL @X(RA),RB TEST
14337 T740 DIV #N,RA TEST ; N:C = 1111
14367 T741 DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
14397 T742 DIV #N,RA TEST ; N:C = 0000 TO 0100
14427 T743 DIV #-N,RA TEST ; RA POS
14450 T744 DIV TEST - V BIT GETS SET
14497 T745 ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14522 T746 ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14548 T747 ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14573 T750 ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14602 T751 ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14632 T752 ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1111 TO 1000
14639 USER CONTROLLED BREAKPOINT -- BIT15
14678 T753 CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
14717 T754 MED TEST - R/W DATA PATTERNS TO REGS
14769 T755 MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14816 T756 MED TEST - CSP CONSTANTS CHECK
14847 T757 MED TEST - MICROBK CHECK OF MICRO-POINTS
14935 T760 PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING

14997 T761 CHECK DISABLE PARITY ERROR TRAP
15068 T762 CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
15106 T763 CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
15184 T764 CHECK ILLEGAL INTERNAL ADDRESS TRAP
15224 T765 CHECK LOG SERVICE & MEMERR LOGS LO-HI BYTE & TAG, IN CACHE ABORT MODE
15347 T766 CHECK "LOG FIRST" MODE OF ERROR LOGGING
15435 T767 CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
15479 END OF PASS ROUTINE
15517 / / / / / / UTILITIES / / / / / /
15520 POWER DOWN AND UP ROUTINES
15574 "T" BIT SERVICE ROUTINE
15581 MICROBREAK TRAP SERVICE ROUTINE
15592 RSVD INSTRUCTION TRAP SERVICE ROUTINE
15653 BUS ERROR TRAP SERVICE ROUTINE
15714 SCOPE HANDLER ROUTINE
15797 ERROR HANDLER ROUTINE
15860 ERROR MESSAGE TYPEOUT ROUTINE
15909 PRINT ROUTINES
15916 TYPE ROUTINE
15996 BINARY TO OCTAL (ASCII) AND TYPE
16074 APT COMMUNICATIONS ROUTINE
16132 TRAP DECODER
16155 TRAP TABLE

```
                                .TITLE  MAINDEC-11-DQKDA-A   KD11-K BASIC LOGIC TESTS
                                ;*COPYRIGHT (C) JULY 1976
                                ;*DIGITAL EQUIPMENT CORP.
                                ;*MAYNARD, MASS. 01754
                                ;*
                                ;*PROGRAM BY JACK RICH
                                ;*
                                ;*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
```

```
                                PR1=    40              ;;PRIORITY LEVEL 1
                                PR2=    100             ;;PRIORITY LEVEL 2
                                PR3=    140             ;;PRIORITY LEVEL 3
                                PR4=    200             ;;PRIORITY LEVEL 4
                                PR5=    240             ;;PRIORITY LEVEL 5
                                PR6=    300             ;;PRIORITY LEVEL 6
                                PR7=    340             ;;PRIORITY LEVEL 7

                                ;*"SWITCH REGISTER" SWITCH DEFINITIONS
                                SW15=   100000
                                SW14=   40000
                                SW13=   20000
                                SW12=   10000
                                SW11=   4000
                                SW10=   2000
                                SW09=   1000
                                SW08=   400
                                SW07=   200
                                SW06=   100
                                SW05=   40
                                SW04=   20
                                SW03=   10
                                SW02=   4
                                SW01=   2
                                SW00=   1
                                .EQUIV  SW09,SW9
                                .EQUIV  SW08,SW8
                                .EQUIV  SW07,SW7
                                .EQUIV  SW06,SW6
                                .EQUIV  SW05,SW5
                                .EQUIV  SW04,SW4
                                .EQUIV  SW03,SW3
                                .EQUIV  SW02,SW2
                                .EQUIV  SW01,SW1
                                .EQUIV  SW00,SW0

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

```
                                .EQUIV  BIT06,BIT6
                                .EQUIV  BIT05,BIT5
                                .EQUIV  BIT04,BIT4
                                .EQUIV  BIT03,BIT3
                                .EQUIV  BIT02,BIT2
                                .EQUIV  BIT01,BIT1
                                .EQUIV  BIT00,BIT0

                                ;*BASIC "CPU" TRAP VECTOR ADDRESSES
000004                          ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
000010                          RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
000014                          TBITVEC=14              ;;"T" BIT
000014                          TRTVEC= 14              ;;TRACE TRAP
000014                          BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
000020                          IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
000024                          PWRVEC= 24              ;;POWER FAIL
000030                          EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
000034                          TRAPVEC=34              ;;"TRAP" TRAP
000060                          TKVEC=  60              ;;TTY KEYBOARD VECTOR
000064                          TPVEC=  64              ;;TTY PRINTER VECTOR
000240                          PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
                                .SBTTL  TRAP CATCHER

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

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

000700                          $APTHD:
000700  000000                $SHIBTS: .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  060570                          $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                  $WR:    .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,U=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  064564                  EM1     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001152  000000                  0
001154  067720                  DT1     ;$REG4,  $REG3,   $REG2, $REG1,$REG0,$ERRPC,$REG5,$REG6
001156  000000                  0
                        ;ITEM 2
001160  064564                  EM2     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001162  065005                  DH2     ;                 IS R3
001164  067720                  DT2     ;$REG4,  $REG3,  $REG2, $REG1, $REG0,$ERRPC,$REG5,$REG6
001166  000000                  0
                        ;ITEM 3
001170  064737                  EM3     ;S/B SP  WAS SP  (IR)  TEST  (PC)  (PSW)
001172  000000                  0
001174  067742                  DT3     ;$REG4,  $REG3,  $REG1,$REG0,$ERRPC,$REG6
001176  000000                  0
                        ;ITEM 4
001200  064564                  EM4     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001202  065016                  DH4     ;                 IS R5
001204  067720                  DT4     ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001206  000000                  0
                        ;ITEM 5
001210  064612                  EM5     ;(IR)  TEST  (PC)  (SP)  (PSW)
001212  000000                  0
001214  067726                  DT5     ;$REG1, $REG0, $ERRPC, $REG5, $REG6
001216  000000                  0
                        ;ITEM 6
001220  064604                  EM6     ;  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001222  000000                  0
001224  067724                  DT6     ;$REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001226  000000                  0
                        ;ITEM 7
001230  064574                  EM7     ;WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001232  000000                  0
001234  067722                  DT7     ;$REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001236  000000                  0
                        ;ITEM 10
001240  064650                  EM10    ;S/B RES  WAS RES  DST OP  STC OP  TEST  (PC) (SP) (PSW)
001242  000000                  0
001244  067720                  DT10    ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001246  000000                  0
```

```
                                ;ITEM 11
001250  065215                          EM11    ;TESTS SKIPPED
001252  065233                          DH11    ;  PC    EXPCTD ACTUAL (TEST #'S)
001254  067760                          DT11    ;$ERRPC, $TESTN,$REGO
001256  000000                          0

                                ;ITEM 12
001260  065271                  EM12    ;MED DID NOT ABORT IN USER MODE
001262  067327                  DH23    ;PC
001264  067620                  DT23    ;$ERRPC
001266  000000                  0

                                ;ITEM 13
001270  065330                  EM13    ;MED EXECUTED IN USER MODE
001272  067327                  DH23    ;PC
001274  067620                  DT23    ;$ERRPC
001276  000000                  0

                                ;ITEM 14
001300  065362                  EM14    ;MED CHANGED PSW
001302  067327                  DH23    ;PC
001304  067620                  DT23    ;$ERRPC
001306  000000                  0

                                ;ITEM 15
001310  065402                  EM15    ;MICROBREAK TRAP-TO-4 DID NOT OCCUR
001312  067240                  DH15    ;ERRPC MEDCODE MICROBK REG.
001314  067564                  DT15    ;$ERRPC,$TMP0,$TMP1,0
001316  067712                  DF15    ;0,0

                                ;ITEM 16

001320  066771                  EM16    ;CACHE DATA LOGGED INCORRECTLY
001322  067543                  DH44    ;PC EXPCT RECVD
001324  067624                  DT24    ;$ERRPC,$REG1,$REG0,0
001326  000000                  0

                                ;ITEM 17

001330  066742                  EM45    ;CACHE TAG LOGGED WRONG
001332  067543                  DH44    ;PC EXPCT RECVD
001334  067624                  DT24    ;$ERRPC,$REG0,$REG1,0
001336  000000                  0

                                ;ITEM 20

001340  065641                  EM26    ;PHYS. BA LOGGED WRONG
001342  067543                  DH44    ;PC EXPCT RECVD
001344  067624                  DT24    ;$ERRPC,$REG1,$REG0,0
001346  000000                  0

                                ;ITEM 21
001350  065471                  EM21    ;CSP CONSTANT WRONG
001352  067272                  DH17    ;PC MEDCODE EXPECTD RECEIVD
001354  067574                  DT21    ;$ERRPC,$TMP1,$TMP2,$REG0,0
001356  067714                  DF17    ;0,0,0
```

```
                                        ;ITEM 22
001360  065514                          EM22    ;BAD DATA READ BY A MED
001362  067272                          DH17    ;PC MEDCODE EXPECTD RECEIVD
001364  067606                          DT22    ;$ERRPC,$TMP1,$TMP2,$TMP3,0
001366  067714                          DF17    ;0,0,0

                                        ;ITEM 23
001370  065543                          EM23    ;NO ODD PC TRAP
001372  067327                          DH23    ;PC
001374  067620                          DT23    ;$ERRPC
001376  000000                          0

                                        ;ITEM 24

001400  065562                          EM24    ;ODD ADR. BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
001402  067334                          DH24    ;PC CPUERR LOGJAM
001404  067624                          DT24    ;$ERRPC,$REG1,$REG0
001406  000000                          0

                                        ;ITEM 25

001410  065445                          EM17    ;LOG CUA LOGGED INCORRECT U-ADDR
001412  067543                          DH44    ;PC     EXPCTD  RECVD
001414  067624                          DT24    ;$ERRPC $REG1   $REG0
001416  000000                          0

                                        ;ITEM 26

001420  065641                          EM26    ;PHYS. BA LOGGED WRONG
001422  067373                          DH26    ;PC PA<17:16>-EXPCT-PA<15:0>  PA<17:16>-RECVD-PA<15:0>
001424  067642                          DT26    ;$ERRPC,$REG1,$REG2,$REG0,$REG3,0
001426  000000                          0

                                        ;ITEM 27

001430  065666                          EM27    ;CACHE PARITY ERROR LOGGED IN BACK UP MODE
001432  067455                          DH27    ;PC LOGPBA LOGDATA LOGTAG
001434  067656                          DT27    ;$ERRPC,$REG3,$REG1,$REG2
001436  000000                          0

                                        ;ITEM 30

001440  065736                          EM30    ;CACHE PARITY TRAPPED WHEN DISABLED
001442  067327                          DH23    ;PC
001444  067620                          DT23    ;$ERRPC
001446  000000                          0

                                        ;ITEM 31

001450  066541                          EM31    ;NO CACHE PARITY TRAP
001452  067327                          DH23    ;PC
001454  067620                          DT23    ;$ERRPC
001456  000000                          0

                                        ;ITEM 32
```

```
                                ;ITEM 33
                                ;ITEM 34
                                ;ITEM 35
                                ;ITEM 36
                                ;ITEM 37
                                ;ITEM 40
                                ;ITEM 41
                                ;ITEM 42
```

```
                                ;ITEM 43
                                ;
001570  066654                  EM43    ;LO BYTE & TAG PARITY BITS NOT SET IN MEM ERR REGISTER
001572  067510                  DH32    ;PC MEMERR
001574  067634                  DT25    ;$ERRPC,$REGO
001576  000000                  0

                                ;ITEM 44

001600  067021                  EMEIS1  ;EIS SET COND CODES WRONG
001602  067156                  DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
001604  067670                  DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REGO   $TMPO
001606  000000                  0

                                ;ITEM 45

001610  067052                  EMEIS2  ;EIS GAVE WRONG RESULT
001612  067156                  DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
001614  067670                  DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REGO   $TMPO
001616  000000                  0

                                ;ITEM 46

001620  067100                  EM46    ;AUTO-INCREMENT (DECREMENT) DID NOT OCCUR
001622  067217                  DH46    ;   PC    (IR)    TEST
001624  067702                  DT46    ;$ERRPC $TMPO   $REGO
001626  000000                  0

        076600                          MED = 076600
        140000                          UM=     140000
        177770                          UBREAK= 177770
        177744                          MEMERR=177744
        177766                          CPUERR=177766
        177746                          CCR=177746
        000100                          WWP=BIT6
        000001                          DPTRP=BIT0
        000200                          PABORT=BIT7
        000100                          LO=BIT6
        000200                          HI=BIT7
        000040                          TAG=BIT5

                                        .EQUIV  SP,KSP


                                ;*      MED OPERATION CODE DEFINITIONS

        000226                          WCNSSW=226
        000022                          RDWHAMI=022
        000222                          WRWHAMI=222
        000144                          RDFLAG=144
```

```
000344                  WRFLAG=344
000100                  RDLJAM=100
000300                  WRLJAM=300
000101                  RDLSERVICE=101
000301                  WRLSERVICE=301
000102                  RDLPBA=102
000302                  WRLPBA=302
000103                  RDLCUA=103
000303                  WRLCUA=303
000104                  RDLFGINT=104
000304                  WRLFGINT=304
000105                  RDLWHAMI=105
000305                  WRLWHAMI=305
000106                  RDLDATA=106
000306                  WRLDATA=306
000107                  RDLTAG=107
000307                  WRLTAG=307
000071                  SWB01=71                ;MICRO ADDR. IN SWAB INST.


                        ;ADDRESS ASSIGNMENTS FOR DL11 CONSOLE TERMINAL INTERFACE

177560                  RCSR=177560             ;RCVR. CONTROL / STATUS REG. ADDRESS
177562                  RDBR = 177562           ;RECEIVER DATA BUFFER REG. ADDR.
177564                  XCSR = 177564           ;TRANSMITTER CONTROL / STATUS REG. ADDR
177566                  XDBR = 177566           ;TRANSMIT DATA BUFFER REG. ADDR.
177546                  LKCSR=  177546          ;LINE CLOCK ADDRESS
```

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

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

```
                                        .SBTTL BTOOS "CCC AND COND. BR'S" TEST - FLAGS CLEARED
                                ; ******************************************************************
                                
001700  000257          BTOOS:  CCC                     ;MAKE N:C=0000
                                
001702  100403          IOOS:   BMI     EOOS            ;BR IF "N" STILL SET
001704  001402                  BEQ     EOOS            ;BR IF "Z" STILL SET
001706  102401                  BVS     EOOS            ;BR IF "V" STILL SET
001710  103002                  BCC     BTOO6           ;BR IF "C" GOT CLEARED
                                
001712  000000          EOOS:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
                                                        ;OR THE CCC FAILED TO CLEAR ALL FLAGS
001714  000771                  BR      BTOOS           ;LOCK ON HARD ERROR
                                
                                ; ******************************************************************
                                        .SBTTL BTOO6 "CLR  %R" TEST - SETS THE "Z" BIT
                                ; ******************************************************************
                                
001716  000257          BTOO6:  CCC                     ;MAKE N:C=0000
                                
001720  005000          IOO6:   CLR     R0              ;TEST THE CLR - IT SHOULD SET "Z"
                                
001722  001402                  BEQ     BTOO7           ;BR IF CLR SET "Z"
                                
001724  000000          EOO6:   HALT                    ;ERROR - CLR FAILED TO SET "Z"
001726  000773                  BR      BTOO6           ;LOCK ON HARD ERROR
                                
                                ; ******************************************************************
                                        .SBTTL BTOO7 "TST  %R" TEST - USING THE CLR
                                ; ******************************************************************
                                
001730  005000          BTOO7:  CLR     R0              ;MAKE [R0] = 000000
001732  000257                  CCC                     ;MAKE N:C=0000
                                
001734  005700          IOO7:   TST     R0              ;TEST THE TST - IT SHOULD SET "Z"
                                
001736  001402                  BEQ     BTO10           ;BR IF "Z" SET OK
                                
001740  000000          EOO7:   HALT                    ;ERROR - CLR FAILED TO LOAD R0 WITH
                                                        ;ALL ZEROES OR TST FAILED
001742  000772                  BR      BTOO7           ;LOCK ON HARD ERROR
                                
                                ; ******************************************************************
                                        .SBTTL BTO10 "COM  %R" TEST - SHOULD SET "N" AND "C"
                                ; ******************************************************************
                                
001744  005000          BTO10:  CLR     R0              ;MAKE [R0] = 000000
001746  000257                  CCC                     ;MAKE N:C=0000
                                
001750  005100          IO10:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
                                
001752  100001                  BPL     EO10            ;BR IF "N" FAILED TO SET
001754  103402                  BCS     BTO11           ;BR IF "C" SET OK
                                
001756  000000          EO10:   HALT                    ;ERROR - COM FAILED
001760  000771                  BR      BTO10           ;LOCK ON HARD ERROR
```

```
                        ; ************************************************************************
                        ;       .SBTTL BTO11 "COM %R AND ADC %R" TEST
                        ; ************************************************************************

001762  005000          BTO11:  CLR     R0              ;MAKE [R0] = 000000
001764  000257                  CCC                     ;MAKE N:C=0000

001766  005100          IO11:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
001770  005500                  ADC     R0              ;TEST THE ADC - [R0] S/B = 000000

001772  001001                  BNE     EO11            ;BR IF "Z" DID NOT SET
001774  103402                  BCS     BTO12           ;BR IF "C" SET OK

001776  000000          EO11:   HALT                    ;ERROR - COM OR ADC FAILED
002000  000770                  BR      BTO11           ;LOCK ON HARD ERROR

                        ; ************************************************************************
                        ;       .SBTTL BTO12 "MOV #N,R" TEST WITH N=177777,[R]=000000
                        ; ************************************************************************

002002  005000          BTO12:  CLR     R0              ;MAKE [R0] = 000000
002004  000257                  CCC                     ;MAKE N:C=0000

002006  012700  177777  IO12:   MOV     #-1,R0          ;TEST THE MOV - [R0] S/B = 177777

002012  005100                  COM     R0              ;MAKE [R0] = 000000
002014  001402                  BEQ     BTO13           ;BR IF "Z" SET

002016  000000          EO12:   HALT                    ;ERROR - MOV FAILED TO LOAD R0 WITH ALL 1'S
002020  000770                  BR      BTO12           ;LOCK ON HARD ERROR

                        ; ************************************************************************
                        ;       .SBTTL BTO13 "MOV #N,R" TEST WITH N=000000,[R]=177777
                        ; ************************************************************************

002022  005000          BTO13:  CLR     R0              ;MAKE [R0] = 000000
002024  005100                  COM     R0              ;MAKE [R0] = 177777
002026  000257                  CCC                     ;SCOPE SYNC

002030  012700  000000  IO13:   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     BTO14           ;BR IF "Z" GOT SET

002042  000000          EO13:   HALT                    ;ERROR - MOV FAILED TO CLEAR R0
002044  000766                  BR      BTO13           ;LOCK ON HARD ERROR

                        ; ************************************************************************
                        ;       .SBTTL BTO14 "CLR (R)" TEST - [R] = 177776
                        ; ************************************************************************

002046  012706  001000  BTO14:  MOV     #STACK,SP       ;SET UP STACK POINTER
002052  012700  177776          MOV     #PSW,R0         ;R0 POINTS TO PSW
002056  000277                  SCC                     ;MAKE [PSW] = 017
```

```
                                ;TEST THE CLR - IT SHOULD CLEAR PSW
002060  005010          I014:   CLR     (R0)
                                ;BR IF CLR MADE "Z" = 0 - IT SHOULD
002062  001002                  BNE     BT015
                                ;ERROR- CLR FAILED TO CLEAR PSW
002064  000000          E014:   HALT
                                ;LOCK ON HARD ERROR
002066  000767                  BR      BT014

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

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

                                ;TEST THE CLR - IT SHOULD CLEAR PSW
002076  005020          I015:   CLR     (R0)+
                                ;BR IF CLR MADE "Z" = 0 - IT SHOULD
002100  001002                  BNE     A015
                                ;ERROR- CLR FAILED TO CLEAR PSW
002102  000000          E1015A: HALT
                                ;LOCK ON HARD ERROR
002104  000771                  BR      BT015

                                ;AUTO INC SHOULD ZERO R0
002106  005700          A015:   TST     R0
                                ;BR IF IT DID
002110  001402                  BEQ     BT016
                                ;ERROR - AUTOINC. FAILED
002112  000000          E2015:  HALT
                                ;LOCK ON HARD ERROR
002114  000765                  BR      BT015

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

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

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

                                ;N:C=1111 ?
002126  100003                  BPL     E016
002130  001002                  BNE     E016
002132  102001                  BVC     E016
002134  103402                  BCS     BT017

                                ;ERROR - COM FAILED TO MAKE [PSW] = 357
002136  000000          E016:   HALT
                                ;LOCK ON HARD ERROR
002140  000766                  BR      BT016

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

                                ;R0 POINTS TO PSW
002142  012700  177776  BT017:  MOV     #PSW,R0
                                ;MAKE [PSW] = 000
002146  005010                  CLR     (R0)
                                ;SCOPE SYNC
002150  000257                  CCC

                                ;TEST THE COM - [PSW] S/B = 357
002152  005120          I017:   COM     (R0)+
```

```
794  002154  100003                    BPL     EA017           ;N:C = 1111 ?
795  002156  001002                    BNE     EA017
796  002160  102001                    BVC     EA017
797  002162  103402                    BCS     A017
798
799  002164  000000            EA017:  HALT                    ;COM FAILED TO SET ALL FLAGS
800  002166  000765                    BR      BT017           ;LOCK ON HARD ERROR
801
802  002170  005100            A017:   COM     R0              ;SHOULD MAKE [R0] = 177777
803  002172  005500                    ADC     R0              ;SHOULD MAKE [R0] = 000000
804  002174  001402                    BEQ     BT020
805
806  002176  000000            E2017:  HALT                    ;ERROR - COM FAILED TO AUTO INC. R0
807  002200  000760                    BR      BT017           ;LOCK ON HARD ERROR
808
809                            ; ******************************************************************************
810                                    .SBTTL BT020 "MOV  RA,RB" TEST - WITH [RA]=177777,[RB]=000000
811                            ; ******************************************************************************
812
813  002202  005000            BT020:  CLR     R0              ;MAKE [R0]=000000
814  002204  005001                    CLR     R1              ;MAKE [R1]=000000
815  002206  005101                    COM     R1              ;MAKE [R1]=0207777
816  002210  000257                    CCC                     ;SCOPE SYNC
817
818  002212  010100            I020:   MOV     R1,R0           ;TEST THE MOV
819
820  002214  100402                    BMI     A020            ;BR IF "N" GOT SET
821
822  002216  000000            EA020:  HALT                    ;ERROR-MOV FAILED TO SET "N"
823  002220  000770                    BR      BT020           ;LOCK ON HARD ERROR
824
825  002222  005100            A020:   COM     R0              ;[R0] SHOULD GO TO 000000
826  002224  001402                    BEQ     BT021           ;BR IF IT DID
827
828  002226  000000            E2020:  HALT                    ;ERROR-MOV FAILED TO LOAD R0 WITH 1'S
829  002230  000764                    BR      BT020           ;LOCK ON HARD ERROR
830
831                            ; ******************************************************************************
832                                    .SBTTL BT021 "MOV  RA,RB" TEST WITH [RA]=000000,[RB]=177777
833                            ; ******************************************************************************
834
835  002232  005000            BT021:  CLR     R0              ;MAKE [R0]=000000
836  002234  005100                    COM     R0              ;MAKE [R0]=177777
837  002236  005001                    CLR     R1              ;MAKE [R1]=000000
838  002240  000257                    CCC                     ;SCOPE SYNC
839
840  002242  010100            I021:   MOV     R1,R0           ;TEST THE MOV
841
842  002244  001402                    BEQ     A021            ;BR IF "Z" GOT SET
843
844  002246  000000            EA021:  HALT                    ;MOV FAILED TO SET "Z"
845  002250  000770                    BR      BT021           ;LOCK ON HARD ERROR
846
847  002252  005100            A021:   COM     R0              ;SHOULD MAKE [R0]=177777 AND SET "C"
848  002254  005500                    ADC     R0              ;SHOULD MAKE [R0]=000000
849  002256  001402                    BEQ     BT022           ;BR IF "Z" SET
```

```
                                ;MOV FAILED TO ZERO R0
002260  000000          E2021:  HALT
002262  000763                  BR      BT021           ;LOCK ON HARD ERROR

                        ; ******************************************************************
                                .SBTTL BT022 "MOV  #N,@#A" TEST WITH N=17,A=177776
                        ; ******************************************************************

002264  000257          BT022:  CCC                     ;MAKE [PSW]=000

002266  012737  000017  177776  I022:   MOV     #17,@#PSW       ;TEST THE MOV

002274  100003                  BPL     E022            ;N:C=1111
002276  001002                  BNE     E022
002300  102001                  BVC     E022
002302  103402                  BCS     BT023

002304  000000          E022:   HALT                    ;MOV FAILED TO LOAD PSW
002306  000766                  BR      BT022           ;LOCK ON HARD ERROR

                        ; ******************************************************************
                                .SBTTL BT023 "MOV  RA,(RB)+" TEST WITH [RA]=17,[RB]=177776
                        ; ******************************************************************

002310  012700  177776  BT023:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002314  012701  000017          MOV     #17,R1          ;[SOURCE]=017
002320  000257                  CCC                     ;SCOPE SYNC - MAKE <N:C> = 0000

002322  010120          I023:   MOV     R1,(R0)+        ;TEST THE MOV

002324  100003                  BPL     EA023           ;N:C = 1111 ?
002326  001002                  BNE     EA023
002330  102001                  BVC     EA023
002332  103402                  BCS     A023

002334  000000          EA023:  HALT                    ;MOV FAILED TO LOAD PSW
002336  000764                  BR      BT023           ;LOCK ON HARD ERROR

002340  005700          A023:   TST     R0              ;DID AUTO INC MAKE R0 GO TO 0?
002342  001402                  BEQ     BT024           ;BR IF IT DID

002344  000000          E2023:  HALT                    ;MOV FAILED TO AUTO INC. R0
002346  000760                  BR      BT023           ;LOCK ON HARD ERROR

                        ; ******************************************************************
                                .SBTTL BT024 "CMP  #N,@#A" TEST WITH N=(A)
                        ; ******************************************************************

002350  012700  177776  BT024:  MOV     #PSW,R0         ;R0 POINTS TO PSW
002354  005010                  CLR     (R0)            ;MAKE [PSW]=000
002356  000273                  273                     ;MAKE N:C=1011

002360  022737  000013  177776  I024:   CMP     #13,@#PSW       ;TEST THE CMP

002366  001402                  BEQ     BT025           ;BR IF "Z" GOT SET
```

```
906  002370  000000                  E024:   HALT                    ;CMP FAILED TO SET "Z"
907  002372  000766                          BR      BT024           ;LOCK ON HARD ERROR
908
909                                  ; ***************************************************************************
910                                          .SBTTL BT025 "CMP  #N,@#A" WITH N >  (A)
911                                  ; ***************************************************************************
912
913  002374  000257                  BT025:  CCC                     ;MAKE [PSW]=000
914
915  002376  022737  000017  177776  I025:   CMP     #17,@#PSW       ;TEST THE CMP
916
917  002404  001401                          BEQ     E025            ;BR IF "Z" GOT SET
918  002406  000402                          BR      BT026           ;GO TO NEXT TEST
919
920  002410  000000                  E025:   HALT                    ;CMP FAILED TO CLEAR "Z"
921  002412  000770                          BR      BT025           ;LOCK ON HARD ERROR
922                                  ; ***************************************************************************
923                                          .SBTTL BT026 "CMP  #N,@#A" WITH N <  (A)
924                                  ; ***************************************************************************
925
926  002414  000277                  BT026:  SCC                     ;MAKE [PSW]=017
927
928  002416  022737  000000  177776  I026:   CMP     #0,@#PSW        ;TEST THE CMP
929
930  002424  001401                          BEQ     E026            ;BR IF "Z" GOT SET
931  002426  000402                          BR      BT027           ;GO TO NEXT TEST
932
933  002430  000000                  E026:   HALT                    ;CMP FAILED TO CLEAR "Z"
934  002432  000770                          BR      BT026           ;LOCK ON HARD ERROR
935
936                                  ; ***************************************************************************
937                                          .SBTTL BT027 "CMP  R,#N" TEST WITH [R]=N
938                                  ; ***************************************************************************
939
940  002434  012700  177777          BT027:  MOV     #-1,R0          ;MAKE [R0]=177777
941  002440  000257                          CCC                     ;N:C=0000
942
943  002442  020027  177777          I027:   CMP     R0,#-1          ;TEST THE CMP
944
945  002446  001402                          BEQ     BT030           ;BR IF CMP SET "Z"
946
947  002450  000000                  E027:   HALT                    ;CMP FAILED
948  002452  000770                          BR      BT027           ;LOCK ON HARD ERROR
949
950                                  ; ***************************************************************************
951                                          .SBTTL BT030 "CMP  R,#N" TEST WITH [R] > N
952                                  ; ***************************************************************************
953
954  002454  012700  000001          BT030:  MOV     #1,R0           ;MAKE [R0]=000001
955  002460  000264                          SEZ                     ;SET THE "Z" BIT
956
957  002462  020027  177777          I030:   CMP     R0,#-1          ;TEST THE CMP
958
959  002466  001002                          BNE     BT031           ;BR IF CMP CLEARED "Z"
960
961  002470  000000                  E030:   HALT                    ;CMP FAILED
```

```
                        002472  000770                          BR      BT030           ;LOCK ON HARD ERROR
                                                        ; ***************************************************************************
                                                                .SBTTL BT031 "CMP  R,#N" TEST WITH [R] < N
                                                        ; ***************************************************************************
                        002474  012700  000001          BT031:  MOV     #1,R0           ;MAKE [R0] = 000001
                        002500  000264                          SEZ                     ;SET THE "Z" BIT
                        002502  020027  000017          I031:   CMP     R0,#17          ;TEST THE CMP
                        002506  001002                          BNE     BT032           ;BR IF CMP CLEARED "Z"
                        002510  000000                  E031:   HALT                    ;CMP FAILED TO SET "Z"
                        002512  000770                          BR      BT031           ;LOCK ON HARD ERROR
                                                        ; ***************************************************************************
                                                                .SBTTL BT032 "CMP  (RA)+,RB" TEST WITH [SOURCE]=[RB]
                                                        ; ***************************************************************************
                        002514  012700  177776          BT032:  MOV     #PSW,R0         ;R0 POINTS TO PSW
                        002520  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW]=340
                        002526  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
                        002532  000257                          CCC                     ;N:C=0000
                        002534  022001                  I032:   CMP     (R0)+,R1        ;TEST THE CMP
                        002536  001402                          BEQ     A032            ;BR IF "Z" GOT SET
                        002540  000000                  EA032:  HALT                    ;CMP FAILED TO ACCESS PSW
                        002542  000764                          BR      BT032           ;LOCK ON HARD ERROR
                        002544  005700                  A032:   TST     R0              ;"Z" SHOULD SET
                        002546  001402                          BEQ     BT033           ;BR IF "Z" SET
                        002550  000000                  E2032:  HALT                    ;CMP FAILED TO AUTO INC. R0
                        002552  000760                          BR      BT032           ;LOCK ON HARD ERROR
                                                        ; ***************************************************************************
                                                                .SBTTL BT033 "CMP  (RA)+,RB" TEST WITH [SOURCE]>[RB]
                                                        ; ***************************************************************************
                        002554  012700  177776          BT033:  MOV     #PSW,R0         ;R0 POINTS TO PSW
                        002560  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW]=340
                        002566  012701  000330                  MOV     #330,R1         ;MAKE [DEST]=330
                        002572  000264                          SEZ                     ;SET THE "Z" BIT
                        002574  022001                  I033:   CMP     (R0)+,R1        ;TEST THE CMP
                        002576  001002                          BNE     A033            ;BR IF "Z" GOT CLEARED
                        002600  000000                  EA033:  HALT                    ;CMP FAILED TO ACCESS PSW
                        002602  000764                          BR      BT033           ;LOCK ON HARD ERROR
                        002604  005700                  A033:   TST     R0              ;"Z" SHOULD SET
                        002606  001402                          BEQ     BT034           ;BR IF "Z" SET
```

```
1018  002610  000000                  E2033:  HALT                    ;CMP FAILED TO AUTO INC. R0
1019  002612  000760                          BR      BT033           ;LOCK ON HARD ERROR
1020                                  ; ***************************************************************
1021                                          .SBTTL BT034 "CMP  (RA)+,RB" TEST WITH [SOURCE]<[RB]
1022                                  ; ***************************************************************
1023
1024  002614  012700  177776          BT034:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1025  002620  012737  000330  177776          MOV     #330,@#PSW      ;MAKE [PSW]=330
1026  002626  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
1027  002632  000264                          SEZ                     ;SET THE "Z" BIT
1028
1029  002634  022001                  I034:   CMP     (R0)+,R1        ;TEST THE CMP
1030
1031  002636  001002                          BNE     A034            ;BR IF "Z" GOT CLEARED
1032
1033  002640  000000                  EA034:  HALT                    ;CMP FAILED TO ACCESS PSW
1034  002642  000764                          BR      BT034           ;LOCK ON HARD ERROR
1035
1036  002644  005700                  A034:   TST     R0              ;"Z" SHOULD SET
1037  002646  001402                          BEQ     BT035           ;BR IF "Z" SET
1038
1039  002650  000000                  E2034:  HALT                    ;CMP FAILED TO AUTO INC. R0
1040  002652  000760                          BR      BT034           ;LOCK ON HARD ERROR
1041                                  ; ***************************************************************
1042                                          .SBTTL BT035 "CMP  RA,RB" TEST WITH [RA] = [RB]
1043                                  ; ***************************************************************
1044
1045  002654  012700  125252          BT035:  MOV     #125252,R0      ;MAKE [R0] = 125252
1046  002660  010001                          MOV     R0,R1           ;MAKE [R1] = 125252
1047  002662  000257                          CCC                     ;SCOPE SYNC
1048
1049  002664  020100                  I035:   CMP     R1,R0           ;TEST THE CMP
1050
1051  002666  001402                          BEQ     BT036           ;BR IF "Z" GOT SET
1052
1053  002670  000000                  E035:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1054  002672  000770                          BR      BT035           ;LOCK ON HARD ERROR
1055                                  ; ***************************************************************
1056                                          .SBTTL BT036 "CMP  RA,RB" TEST WITH [RA] < [RB]
1057                                  ; ***************************************************************
1058
1059  002674  012700  025252          BT036:  MOV     #25252,R0       ;MAKE [R0] = 25252
1060  002700  005001                          CLR     R1              ;MAKE [R1] = 000000
1061  002702  000264                          SEZ                     ;SCOPE SYNC - SET "Z"
1062
1063  002704  020100                  I036:   CMP     R1,R0           ;TEST THE CMP
1064
1065  002706  001002                          BNE     BT037           ;BR IF "Z" GOT  CLEARED
1066
1067  002710  000000                  E036:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1068  002712  000770                          BR      BT036           ;LOCK ON HARD ERROR
1069                                  ; ***************************************************************
1070                                          .SBTTL BT037 "CMP  RA,RB" TEST WITH [RA] > [RB]
1071                                  ; ***************************************************************
1072
1073  002714  005000                  BT037:  CLR     R0              ;MAKE [R0] = 000000
```

```
1074  002716  012701  000017          MOV     #17,R1          ;MAKE [R1] = 000017
1075  002722  000264                  SEZ                     ;SCOPE SYNC - SET "Z"
1076
1077  002724  020100          I037:   CMP     R1,R0           ;TEST THE CMP
1078
1079  002726  001002                  BNE     BT040           ;BR IF "Z" GOT CLEARED
1080
1081  002730  000000          E037:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1082  002732  000770                  BR      BT037           ;LOCK ON HARD ERROR
1083
1084                          ; ***************************************************************
1085                                  .SBTTL BT040 "MOV  (RA),RB" TEST WITH [SOURCE]=[RB]=17
1086                          ; ***************************************************************
1087
1088  002734  012700  177776  BT040:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1089  002740  005010                  CLR     (R0)            ;MAKE [PSW]=000
1090  002742  005001                  CLR     R1              ;MAKE [R1]=000000
1091  002744  000277                  SCC                     ;MAKE N:C=1111
1092
1093  002746  011001          I040:   MOV     (R0),R1         ;TEST THE MOV
1094
1095  002750  020127  000017          CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1096  002754  001402                  BEQ     BT041           ;BR IF YES
1097
1098  002756  000000          E040:   HALT                    ;MOV FAILED TO LOAD R1
1099  002760  000765                  BR      BT040           ;LOCK ON HARD ERROR
1100                          ; ***************************************************************
1101                                  .SBTTL BT041 "MOV  (RA)+,RB" TEST WITH [SOURCE]=[RB]=17
1102                          ; ***************************************************************
1103
1104  002762  012700  177776  BT041:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1105  002766  005010                  CLR     (R0)            ;MAKE [PSW]=000
1106  002770  005001                  CLR     R1              ;MAKE [R1]=000000
1107  002772  000277                  SCC                     ;MAKE N:C=1111
1108
1109  002774  012001          I041:   MOV     (R0)+,R1                ;TEST THE MOV
1110
1111  002776  020127  000017          CMP     R1,#17          ;DID R1 GET LOADED WITH 000017 ?
1112  003002  001402                  BEQ     A041            ;BR IF YES
1113
1114  003004  000000          EA041:  HALT                    ;MOV FAILED TO LOAD R1
1115  003006  000765                  BR      BT041           ;LOCK ON HARD ERROR
1116
1117  003010  005700          A041:   TST     R0              ;"Z" SHOULD SET
1118  003012  001402                  BEQ     BT042           ;BR IF "Z" GOT SET
1119
1120  003014  000000          E2041:  HALT                    ;MOV FAILED TO AUTO INC. R0
1121  003016  000761                  BR      BT041           ;LOCK ON HARD ERROR
1122
1123                          ; ***************************************************************
1124                                  .SBTTL BT042 "XOR  RA,RB" TEST WITH [RA] = [RB] = 000000
1125                          ; ***************************************************************
1126
1127  003020  005000          BT042:  CLR     R0              ;MAKE [R0] = 000000
1128  003022  005001                  CLR     R1              ;MAKE [R1] = 000000
1129  003024  000257                  CCC                     ;SCOPE SYNC
```

```
1130
1131  003026  074100                  I042:   XOR     R1,R0           ;TEST THE XOR
1132
1133  003030  005700                          TST     R0              ;RESULT = 000000 ?
1134  003032  001402                          BEQ     BT043           ;BR IF YES
1135
1136  003034  000000                  E042:   HALT                    ;XOR FAILED
1137  003036  000770                          BR      BT042
1138
1139                                  ; ***************************************************************************
1140                                          .SBTTL BT043 "XOR  RA,RB" TEST WITH [RA] = [RB] = 177777
1141                                  ; ***************************************************************************
1142
1143  003040  005000                  BT043:  CLR     R0              ;MAKE [R0] = 177777
1144  003042  005100                          COM     R0
1145  003044  010001                          MOV     R0,R1           ;MAKE [R1] = 177777
1146  003046  000257                          CCC                     ;SCOPE SYNC
1147
1148  003050  074100                  I043:   XOR     R1,R0           ;TEST THE XOR
1149
1150  003052  005700                          TST     R0              ;RESULT = 000000 ?
1151  003054  001402                          BEQ     BT044   ;BR IF YES
1152
1153  003056  000000                  E043:   HALT                    ;XOR FAILED
1154  003060  000767                          BR      BT043           ;LOCK ON HARD ERROR
1155
1156                                  ; ***************************************************************************
1157                                          .SBTTL BT044 "XOR  RA,RB" TEST WITH [RB]=052525,[RA]=125252
1158                                  ; ***************************************************************************
1159
1160  003062  012701  125252          BT044:  MOV     #125252,R1      ;MAKE [R1]=125252
1161  003066  012700  052525                  MOV     #052525,R0      ;MAKE [R0]=052525
1162  003072  000257                          CCC                     ;SCOPE SYNC
1163
1164  003074  074100                  I044:   XOR     R1,R0           ;TEST THE XOR
1165
1166  003076  020027  177777                  CMP     R0,#-1          ;RESULT = 177777 ?
1167  003102  001402                          BEQ     BT045           ;BR IF YES
1168
1169  003104  000000                  E044:   HALT                    ;XOR FAILED
1170  003106  000400                          BR      BT045           ;LOCK ON HARD ERROR
1171                                  ; ***************************************************************************
1172                                          .SBTTL BT045 "XOR  RA,RB" TEST WITH [RA]=052525,[RB]=125252
1173                                  ; ***************************************************************************
1174
1175  003110  012700  125252          BT045:  MOV     #125252,R0      ;MAKE [R0]=125252
1176  003114  012701  052525                  MOV     #052525,R1      ;MAKE [R1]=052525
1177  003120  000257                          CCC                     ;SCOPE SYNC
1178
1179  003122  074100                  I045:   XOR     R1,R0           ;TEST THE XOR
1180
1181  003124  020027  177777                  CMP     R0,#-1          ;RESULT = 177777 ?
1182  003130  001402                          BEQ     BT046           ;BR IF YES
1183
1184  003132  000000                  E045:   HALT                    ;XOR FAILED
1185  003134  000765                          BR      BT045           ;LOCK ON HARD ERROR
```

```
1186
1187                                    ; ***************************************************************************
1188                                            .SBTTL BT046 GPR ADDRESS INTERRACTION TEST
1189                                    ; ***************************************************************************
1190
1191  003136  012700  125252    BT046:  MOV     #125252,R0      ;[R0] = 125252
1192  003142  010001                    MOV     R0,R1
1193  003144  005101                    COM     R1              ;[R1] = 052525
1194  003146  010102                    MOV     R1,R2
1195  003150  005102                    COM     R2              ;[R2] = 125252
1196  003152  010203                    MOV     R2,R3
1197  003154  005103                    COM     R3              ;[R3] = 052525
1198  003156  010304                    MOV     R3,R4
1199  003160  005104                    COM     R4              ;[R4] = 125252
1200  003162  010405                    MOV     R4,R5
1201  003164  005105                    COM     R5              ;[R5] = 052525
1202
1203  003166  074100            I046:   XOR     R1,R0           ;[R0] S/B = 177777
1204  003170  074200                    XOR     R2,R0           ;[R0] S/B = 125252
1205  003172  074300                    XOR     R3,R0           ;[R0] S/B = 177777
1206  003174  074400                    XOR     R4,R0           ;[R0]  S/B = 125252
1207  003176  074500                    XOR     R5,R0           ;[R0]  S/B = 177777
1208  003200  005100                    COM     R0              ;[R0] S/B = 000000
1209
1210  003202  001402                    BEQ     A046            ;BR IF [R0] WAS 000000
1211
1212  003204  000000            EA046:  HALT                    ;GPR ADDRESSING PROBLEM
1213  003206  000753                    BR      BT046           ;LOCK ON HARD ERROR
1214
1215  003210  020627  001000    A046:   CMP     SP,#STACK       ;DID R6 GET DISTURBED
1216  003214  001402                    BEQ     BASIC           ;BR IF NOT
1217
1218  003216  000000            E2046:  HALT                    ;R6 ADDRESS PROBLEM
1219  003220  000746                    BR      BT046           ;LOCK ON HARD ERROR
```

```
1220                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1221                                    ;          /  /  /  /  /  / BASIC INSTRUCTION TESTS /  /  /  /  /  /
1222                                    ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1223
1224  003222  005037  063200            BASIC:  CLR     @#ONCE          ;SIGNAL PROGRAM HEADER TO BE PRINTED
1225  003226  005037  001012                    CLR     @#$ERTTL        ;CLEAR ERROR COUNT FIRST TIME THROUGH
1226  003232  005037  001126                    CLR     @#$PASS         ;CLEAR PASS COUNT FIRST TIME THROUGH
1227  003236  012701  063162            INIT:   MOV     #PRIFLG,R1      ;SET UP TO INIT. COUNTERS AND FLAGS
1228  003242  005021                    1$:     CLR     (R1)+           ;CLEAR ONE WORD
1229  003244  020127  063200                    CMP     R1,#ONCE        ;CLEARED ALL FLAGS AND COUNTERS?
1230  003250  001374                            BNE     1$              ;BR IF NOT
1231  003252  012706  001000                    MOV     #STACK,SP       ;SET UP THE STACK POINTER
1232  003256  012737  004030  177770            MOV     #4030,@#UBREAK  ;SET SCOPE SYNC FOR COND CODE OPERATE
1233  003264  012737  177777  001074            MOV     #-1,@#$REGS     ;FLAG CURRENT STACK POINTER TO BE TYPED
1234                                                                    ;IN FIRST ERROR CALL
1235
1236                                    ;;**********************************************************************
1237                                    ;*TEST 0         BASIC "BNE" TEST WITH Z=0
1238                                    ;;**********************************************************************
1239  003272                            TST0:
1240  003272  012700  000000                    MOV     #0,R0           ;;LOAD R0 WITH TEST NUMBER
1241  003276  000257                    1$:     CCC                     ;MAKE Z=0
1242
1243  003300                            2$:
1244  003300  001002                            BNE     TST1            ;;TEST THE BNE - IT SHOULD BR
1245
1246  003302  000000                    3$:     HALT                    ;BNE FAILED TO LOAD PC
1247  003304  000774                            BR      1$              ;LOCK ON HARD ERROR
1248
1249                                    ;;**********************************************************************
1250                                    ;*TEST 1         BASIC "BNE" TEST WITH Z=1
1251                                    ;;**********************************************************************
1252  003306                            TST1:
1253  003306  012700  000001                    MOV     #1,R0           ;;LOAD R0 WITH TEST NUMBER
1254  003312  000264                    1$:     SEZ                     ;SET THE "Z" BIT
1255
1256  003314  001001                    2$:     BNE     3$              ;TEST THE BNE - IT SHOULD NOT BR
1257
1258  003316  000402                            BR      TST2            ;;GO TO NEXT TEST
1259
1260  003320  000000                    3$:     HALT                    ;BNE BRANCHED WITH Z=1
1261  003322  000773                            BR      1$              ;LOCK ON HARD ERROR
1262
1263                                    ;;**********************************************************************
1264                                    ;*TEST 2         BASIC "BEQ" TEST WITH Z=1
1265                                    ;;**********************************************************************
1266  003324                            TST2:
1267  003324  012700  000002                    MOV     #2,R0           ;;LOAD R0 WITH TEST NUMBER
1268  003330  000264                    1$:     SEZ                     ;MAKE Z=1
1269
1270  003332                            2$:
1271  003332  001402                            BEQ     TST3            ;;TEST THE BEQ - IT SHOULD BR
1272
1273  003334  000000                    3$:     HALT                    ;BEQ FAILED TO LOAD THE PC
1274  003336  000774                            BR      1$              ;LOCK ON HARD ERROR
1275                                    ;;**********************************************************************
```

```
                                ;*TEST 3        BASIC "BEQ" TEST WITH Z=0
                                ;;**************************************************************
003340                          TST3:
003340  012700  000003                  MOV     #3,R0           ;;LOAD R0 WITH TEST NUMBER
003344  000257                  1$:     CCC                     ;MAKE Z=0

003346  001401                  2$:     BEQ     3$              ;TEST THE BEQ - IT SHOULD NOT BR

003350  000402                          BR      TST4            ;;GO TO NEXT TEST

003352  000000                  3$:     HALT                    ;BEQ BRANCHED WITH Z=0
003354  000773                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 4        BASIC "BPL" TEST WITH N=1
                                ;;**************************************************************
003356                          TST4:
003356  012700  000004                  MOV     #4,R0           ;;LOAD R0 WITH TEST NUMBER
003362  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003366  000270                          SEN                     ;MAKE N=1

003370  100001                  2$:     BPL     3$              ;TEST THE BPL - IT SHOULDN'T BR

003372  000402                          BR      TST5            ;;GO TO NEXT TEST

003374  000000                  3$:     HALT                    ;BPL BRANCHED WITH N=1
003376  000771                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 5        BASIC "BPL" TEST WITH N=0
                                ;;**************************************************************
003400                          TST5:
003400  012700  000005                  MOV     #5,R0           ;;LOAD R0 WITH TEST NUMBER
003404  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003410  000257                          CCC                     ;SCOPE SYNC

003412                          2$:
003412  100002                          BPL     TST6            ;;TEST THE BPL - IT SHOULD BR

003414  000000                  3$:     HALT                    ;BPL FAILED TO LOAD THE PC
003416  000772                          BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 6        BASIC "MOV  (R4),R8" TEST - (R4)=177776
                                ;;**************************************************************
003420                          TST6:
003420  012700  000006                  MOV     #6,R0           ;;LOAD R0 WITH TEST NUMBER
003424  012705  177776                  MOV     #PSW,R5         ;SOURCE ADDR = 177776
003430  005015                  1$:     CLR     (R5)            ;MAKE [PSW]=000
003432  005003                          CLR     R3              ;[DEST] = 000000
003434  000277                          SCC                     ;MAKE [PSW]=017

003436  011503                  2$:     MOV     (R5),R3         ;TEST THE MOV

003440  020327  000017                  CMP     R3,#17          ;CORRECT RESULT ?
003444  001402                          BEQ     TST7            ;;BR IF YES
```

```
1332
1333  003446  000000                  3$:     HALT                    ;ERROR-MOV FAILED
1334  003450  000767                          BR      1$              ;LOCK ON HARD ERROR
1335                                  ;;*********************************************************************
1336                                  ;*TEST 7       BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
1337                                  ;;*********************************************************************
1338  003452                          TST7:
1339  003452  012700  000007                 MOV     #7,R0           ;;LOAD R0 WITH TEST NUMBER
1340  003456  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1341  003462  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
1342  003466  012737  125252  063236 1$:     MOV     #125252,@#MBUF0 ;MAKE [DEST] = 125252
1343  003474  000257                         CCC                     ;MAKE N:C=0000
1344
1345  003476  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
1346
1347  003500  001402                          BEQ     TST10           ;;BR IF "Z" GOT SET
1348
1349  003502  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO SET "Z"
1350  003504  000770                          BR      1$              ;LOCK ON HARD ERROR
1351                                  ;;*********************************************************************
1352                                  ;*TEST 10      BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
1353                                  ;;*********************************************************************
1354  003506                          TST10:
1355  003506  012700  000010                 MOV     #10,R0          ;;LOAD R0 WITH TEST NUMBER
1356  003512  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1357  003516  012704  000001                 MOV     #1,R4           ;RESULT S / B = 000001
1358  003522  005037  063236          1$:    CLR     @#MBUF0         ;MAKE [DEST] = 000000
1359  003526  000264                         SEZ                     ;MAKE N:C=0100
1360
1361  003530  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
1362
1363  003532  001002                          BNE     TST11           ;;BR IF "Z" GOT CLEARED
1364
1365  003534  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO CLR "Z"
1366  003536  000771                          BR      1$              ;LOCK ON HARD ERROR
1367
1368                                  ;;*********************************************************************
1369                                  ;*TEST 11      BASIC "CMP #N,R" TEST - N = [R]
1370                                  ;;*********************************************************************
1371  003540                          TST11:
1372  003540  012700  000011                 MOV     #11,R0          ;;LOAD R0 WITH TEST NUMBER
1373  003544  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
1374  003550  010403                  1$:    MOV     R4,R3           ;[DEST] = 125252
1375  003552  000257                         CCC                     ;SCOPE SYNC
1376
1377  003554  022703  125252          2$:    CMP     #125252,R3      ;TEST THE CMP
1378
1379  003560  001402                          BEQ     4$              ;BR IF N = [R]
1380
1381  003562  000000                  3$:     HALT                    ;CMP FAILED
1382  003564  000771                          BR      1$              ;LOCK ON HARD ERROR
1383
1384  003566  020403                  4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
1385  003570  001402                          BEQ     TST12           ;;BR IF NO
1386
1387  003572  000000                  5$:     HALT                    ;CMP DELIVERED A RESULT
```

```
1388  003574  000765                    BR      1$              ;LOCK ON HARC ERROR
1389
1390                                ;;*****************************************************************
1391                                ;*TEST 12       BASIC "CMP #N,R" TEST - N NOT EQUAL TO [R]
1392                                ;;*****************************************************************
1393  003576                        TST12:
1394  003576  012700  000012               MOV     #12,R0          ;;LOAD R0 WITH TEST NUMBER
1395  003602  005004                       CLR     R4              ;RESULT S / B = 000000
1396  003604  010403                1$:    MOV     R4,R3           ;[DEST] = 125252
1397  003606  000264                       SEZ                     ;SCOPE SYNC
1398
1399  003610  022703  000001        2$:    CMP     #1,R3           ;TEST THE CMP
1400
1401  003614  001002                       BNE     4$              ;BR IF N NOT EQUAL TO [R]
1402
1403  003616  000000                3$:    HALT                    ;CMP FAILED
1404  003620  000771                       BR      1$              ;LOCK ON HARD ERROR
1405
1406  003622  020403                4$:    CMP     R4,R3           ;DID CMP ALTER [DEST]?
1407  003624  001402                       BEQ     TST13           ;;BR IF NO
1408
1409  003626  000000                5$:    HALT                    ;CMP DELIVERED A RESULT
1410  003630  000765                       BR      1$              ;LOCK ON HARD ERROR
1411
1412                                ;;*****************************************************************
1413                                ;*TEST 13       BASIC "MOV  RA,(RB)" TEST
1414                                ;;*****************************************************************
1415  003632                        TST13:
1416  003632  012700  000013               MOV     #13,R0          ;;LOAD R0 WITH TEST NUMBER
1417  003636  012702  063236               MOV     #MBUF0,R2       ;DEST ADDR=MBUF0
1418  003642  012704  177777               MOV     #-1,R4          ;RESULT S / B = 177777
1419  003646  005012                1$:    CLR     (R2)            ;MAKE [DEST] = 000000
1420  003650  000257                       CCC                     ;SCOPE SYNC - N:C=0000
1421
1422  003652  010412                2$:    MOV     R4,(R2)         ;TEST THE MOV
1423
1424  003654  020412                       CMP     R4,(R2)         ;RESULT CORRECT ?
1425  003656  001402                       BEQ     TST14           ;;BR IF  YES
1426
1427  003660  000000                3$:    HALT                    ;ERROR - MOV FAILED
1428  003662  000771                       BR      1$              ;LOCK ON HARD ERROR
1429
1430                                ;;*****************************************************************
1431                                ;*TEST 14       BASIC "MOV  #N,(R)" TEST
1432                                ;;*****************************************************************
1433  003664                        TST14:
1434  003664  012700  000014               MOV     #14,R0          ;;LOAD R0 WITH TEST NUMBER
1435  003670  012702  063236               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1436  003674  012704  177777               MOV     #-1,R4          ;RESULT S / B = 177777
1437  003700  005012                1$:    CLR     (R2)            ;MAKE [DEST] = 000000
1438  003702  000257                       CCC                     ;SCOPE SYNC
1439
1440  003704  012712  177777        2$:    MOV     #-1,(R2)        ;TEST THE MOV
1441
1442  003710  020412                       CMP     R4,(R2)         ;RESULT OK ?
1443  003712  001402                       BEQ     TST15           ;;BR IF YES
```

```
1444
1445  003714  000000                  3$:     HALT                    ;ERROR - MOV FAILED
1446  003716  000770                          BR      1$              ;LOCK ON HARD ERROR
1447
1448                                  ;;*****************************************************************
1449                                  ;*TEST 15       BASIC "MOVB #N,X(R)" TEST - DEST EVEN
1450                                  ;;*****************************************************************
1451  003720                          TST15:
1452  003720  012700  000015                 MOV     #15,R0          ;;LOAD R0 WITH TEST NUMBER
1453  003724  012704  177401                 MOV     #177401,R4      ;RESULT S / B = 177401
1454  003730  012702  063242                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
1455  003734  012705  063236                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
1456  003740  012712  177777         1$:     MOV     #-1,(R2)        ;[DEST] = 177777
1457  003744  000257                          CCC                     ;SCOPE SYNC
1458
1459  003746  112765  000001  000004  2$:     MOVB    #1,4(R5)        ;TEST THE MOVB
1460
1461  003754  020412                          CMP     R4,(R2)         ;RESULT OK?
1462  003756  001402                          BEQ     TST16           ;;BR IF YES
1463
1464  003760  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
1465  003762  000766                          BR      1$              ;LOCK ON HARD ERROR
1466
1467                                  ;;*****************************************************************
1468                                  ;*TEST 16       BASIC "MOVB #N,X(R)" TEST - DEST ODD
1469                                  ;;*****************************************************************
1470  003764                          TST16:
1471  003764  012700  000016                 MOV     #16,R0          ;;LOAD R0 WITH TEST NUMBER
1472  003770  012704  000777                 MOV     #777,R4         ;RESULT S / B = 777
1473  003774  012702  063242                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
1474  004000  012705  063236                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
1475  004004  012712  177777         1$:     MOV     #-1,(R2)        ;[DEST] = 177777
1476  004010  000257                          CCC                     ;SCOPE SYNC
1477
1478  004012  112765  000001  000005  2$:     MOVB    #1,5(R5)        ;TEST THE MOVB
1479
1480  004020  020412                          CMP     R4,(R2)         ;RESULT OK?
1481  004022  001402                          BEQ     TST17           ;;BR IF YES
1482
1483  004024  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
1484  004026  000766                          BR      1$              ;LOCK ON HARD ERROR
1485
1486                                  ;;*****************************************************************
1487                                  ;*TEST 17       BASIC "TST  @#A" TEST WITH [A] GT 0
1488                                  ;;*****************************************************************
1489  004030                          TST17:
1490  004030  012700  000017                 MOV     #17,R0          ;;LOAD R0 WITH TEST NUMBER
1491  004034  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1492  004040  012704  000377                 MOV     #377,R4         ;RESULT S / B = 377 (NO CHANGE)
1493  004044  010412           1$:     MOV     R4,(R2)         ;[DEST] = 377
1494  004046  000257                          CCC                     ;SCOPE SYNC
1495
1496  004050  005737  063236          2$:     TST     @#MBUF0         ;TEST THE TST
1497
1498  004054  001401                          BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
1499  004056  100002                          BPL     TST20           ;;BR IF "N" CLEAR - IT SHOULD BE
```

```
1500
1501  004060  000000                  3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
1502  004062  000770                          BR      1$              ;LOCK ON HARD ERROR
1503                                  ;;**********************************************************************
1504                                  ;*TEST 20       BASIC "TST  @#A" TEST WITH [A] LT 0
1505                                  ;;**********************************************************************
1506  004064                          TST20:
1507  004064  012700  000020                MOV     #20,R0          ;;LOAD R0 WITH TEST NUMBER
1508  004070  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1509  004074  012704  100000                MOV     #100000,R4      ;MAKE S / B = 100000
1510  004100  010412                  1$:   MOV     R4,(R2)         ;MAKE [DEST] = 100000
1511  004102  000257                        CCC                     ;SCOPE SYNC
1512
1513  004104  005737  063236          2$:   TST     @#MBUF0         ;TEST THE TST
1514
1515  004110  001401                        BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
1516  004112  100402                        BMI     4$              ;BR IF "N" SET - IT SHOULD BE
1517
1518  004114  000000                  3$:   HALT                    ;TST FAILED TO ALTER CODES PROPERLY
1519  004116  000770                        BR      1$              ;LOCK ON HARD ERROR
1520  004120  020412                  4$:   CMP     R4,(R2)         ;DID TST DISTURB [DEST] ?
1521  004122  001402                        BEQ     TST21           ;;BR IF NOT
1522
1523  004124  000000                  5$:   HALT                    ;TST DELIVERED A RESULT
1524  004126  000764                        BR      1$              ;LOCK ON HARD ERROR
1525
1526                                  ;;**********************************************************************
1527                                  ;*TEST 21       BASIC "TST  @#A" WITH [A] = 0
1528                                  ;;**********************************************************************
1529  004130                          TST21:
1530  004130  012700  000021                MOV     #21,R0          ;;LOAD R0 WITH TEST NUMBER
1531  004134  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1532  004140  005004                        CLR     R4              ;RESULT S / B = 0 (IT SHOULDN'T CHANGE
1533  004142  005012                  1$:   CLR     (R2)            ;[DEST] = 0
1534  004144  000257                        CCC                     ;SCOPE SYNC - Z=0
1535
1536  004146  005737  063236          2$:   TST     @#MBUF0         ;TEST THE TST
1537
1538  004152  001402                        BEQ     4$              ;BR IF TST SET "Z"
1539
1540  004154  000000                  3$:   HALT                    ;TST FAILED TO SET "Z"
1541  004156  000771                        BR      1$              ;LOCK ON HARD ERROR
1542
1543  004160  020412                  4$:   CMP     R4,(R2)         ;[DEST] STILL = 000000
1544  004162  001402                        BEQ     TST22           ;;BR IF YES
1545
1546  004164  000000                  5$:   HALT                    ;TST ALTERED THE [DEST]
1547  004166  000765                        BR      1$              ;LOCK ON HARD ERROR
1548
1549                                  ;;**********************************************************************
1550                                  ;*TEST 22       BASIC "BIT  #N,@#A" WITH BIT SET IN "A"
1551                                  ;;**********************************************************************
1552  004170                          TST22:
1553  004170  012700  000022                MOV     #22,R0          ;;LOAD R0 WITH TEST NUMBER
1554  004174  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1555  004200  012704  040000                MOV     #40000,R4       ;RESULT S / B = 40000
```

```
1556  004204  010412                  1$:     MOV     R4,(R2)         ;MAKE [DEST] = 40000
1557  004206  000277                          SCC                     ;SCOPE SYNC - Z=1
1558
1559  004210  032737  040000  063236  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
1560
1561  004216  001002                          BNE     TST23           ;;BR IF Z=0 - IT SHOULD BE
1562
1563  004220  000000                  3$:     HALT                    ;BIT FAILED TO CLEAR "Z"
1564  004222  000770                          BR      1$              ;LOCK ON HARD ERROR
1565
1566                                  ;;**************************************************************
1567                                  ;*TEST 23       BASIC "BIT  #N,@#A" WITH BIT CLEAR IN "A"
1568                                  ;;**************************************************************
1569  004224                          TST23:
1570  004224  012700  000023                  MOV     #23,R0          ;;LOAD R0 WITH TEST NUMBER
1571  004230  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1572  004234  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
1573  004236  000257                          CCC                     ;SCOPE SYNC - Z=0
1574
1575  004240  032737  040000  063236  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
1576
1577  004246  001402                          BEQ     4$              ;BR IF Z=1 - IT SHOULD BE
1578
1579  004250  000000                  3$:     HALT                    ;BIT FAILED TO SET "Z"
1580  004252  000770                          BR      1$              ;LOCK ON HARD ERROR
1581
1582  004254  005712                  4$:     TST     (R2)            ;DID BIT DELIVER A RESULT
1583  004256  001402                          BEQ     TST24           ;;BR IF NOT
1584
1585  004260  000000                  5$:     HALT                    ;BIT DISTURBED THE [DEST]
1586  004262  000764                          BR      1$              ;LOCK ON HARD ERROR
1587
1588                                  ;;**************************************************************
1589                                  ;*TEST 24       BASIC "TST  (R)+" TEST
1590                                  ;;**************************************************************
1591  004264                          TST24:
1592  004264  012700  000024                  MOV     #24,R0          ;;LOAD R0 WITH TEST NUMBER
1593                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT0
1594  004270  032737  000001  063160          BIT     #BIT0,@#BPTLOC  ;BREAKPOINT HALT SET ??
1595  004276  001401                          BEQ     .+4             ;BR IF NOT
1596  004300  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
1597  004302  012702  063236                  MOV     #MBUF0,R2       ;INITIAL DEST ADDR = MBUF0
1598  004306  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
1599  004310  000257                          CCC                     ;SCOPE SYNC
1600
1601  004312  005722                  2$:     TST     (R2)+           ;TEST THE TST
1602
1603  004314  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
1604
1605  004316  000000                  3$:     HALT                    ;TST FAILED TO SET "Z"
1606  004320  000772                          BR      1$              ;LOCK ON HARD ERROR
1607
1608  004322  022702  063240          4$:     CMP     #MBUF0+2,R2     ;DID REG. GET AUTO-INCREMENTED ?
1609  004326  001402                          BEQ     TST25           ;;BR IF YES
1610
1611  004330  000000                  5$:     HALT                    ;TST FAILED TO UPDATE REGISTER
```

```
1612  004332  000765                        BR      1$              ;LOCK ON HARD ERROR
1613
1614                                  ;;*****************************************************************
1615                                  ;*TEST 25        BASIC "TST -(R)" TEST
1616                                  ;;*****************************************************************
1617  004334                          TST25:
1618  004334  012700  000025                MOV     #25,R0          ;;LOAD R0 WITH TEST NUMBER
1619  004340  012702  063254                MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
1620  004344  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
1621  004350  012705  063256        1$:     MOV     #DWTA+10,R5     ;BASE DEST ADDR = DWTA+10
1622  004354  000270                        SEN                     ;SCOPE SYNC
1623
1624  004356  005745                2$:     TST     -(R5)           ;TEST THE TST
1625
1626  004360  100002                        BPL     4$              ;BR IF "N" CLEAR
1627
1628  004362  000000                3$:     HALT                    ;TST FAILED TO CLEAR "N"
1629  004364  000771                        BR      1$              ;LOCK ON HARD ERROR
1630
1631  004366  020502                4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED?
1632  004370  001402                        BEQ     6$              ;BR IF YES
1633
1634  004372  000000                5$:     HALT                    ;ERROR - TST FAILED TO UPDATE DEST REG
1635  004374  000765                        BR      1$              ;LOCK ON HARD ERROR
1636
1637  004376  020412                6$:     CMP     R4,(R2)         ;DID TST ALTER [DEST]?
1638  004400  001403                        BEQ     TST26           ;;BR IF NOT
1639
1F    004402  000000                7$:     HALT                    ;TST ALTERED [DEST]
1     004404  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
1. 2  004406  000760                        BR      1$              ;LOCK ON HARD ERROR
1643
1644                                  ;;*****************************************************************
1645                                  ;*TEST 26        BASIC "COM  @#A" TEST
1646                                  ;;*****************************************************************
1647  004410                          TST26:
1648  004410  012700  000026                MOV     #26,R0          ;;LOAD R0 WITH TEST NUMBER
1649  004414  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1650  004420  005004                        CLR     R4              ;RESULT S / B = 177777
1651  004422  005104                        COM     R4
1652  004424  005012                1$:     CLR     (R2)            ;MAKE [DEST] = 000000
1653  004426  000257                        CCC                     ;SCOPE SYNC
1654
1655  004430  005137  063236        2$:     COM     @#MBUF0 ;TEST THE COM
1656
1657  004434  020412                        CMP     R4,(R2)         ;RESULT = 177777 ??
1658  004436  001402                        BEQ     TST27           ;;BR IF YES
1659
1660  004440  000000                3$:     HALT                    ;COM DELIVERED THE WRONG RESULT
1661  004442  000770                        BR      1$
1662
1663                                  ;;*****************************************************************
1664                                  ;*TEST 27        BASIC "INC  @#A" TEST
1665                                  ;;*****************************************************************
1666  004444                          TST27:
1667  004444  012700  000027                MOV     #27,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
1668  004450  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1669  004454  012704  000100          MOV     #100,R4         ;RESULT S / B = 100
1670  004460  012712  000077  1$:     MOV     #77,(R2)        ;[DEST] = 77
1671  004464  000257                  CCC                     ;SCOPE SYNC
1672
1673  004466  005237  063236  2$:     INC     @#MBUF0         ;TEST THE INC
1674
1675  004472  020412                  CMP     R4,(R2)         ;DID RESULT = 100 ??
1676  004474  001402                  BEQ     TST30           ;;BR IF YES
1677
1678  004476  000000  3$:     HALT                            ;INC DELIVERED WRONG RESULT
1679  004500  000767                  BR      1$              ;LOCK ON HARD ERROR
1680
1681                          ;;*****************************************************************
1682                          ;*TEST 30        BASIC "DEC  RN" TEST
1683                          ;;*****************************************************************
1684  004502                  TST30:
1685  004502  012700  000030          MOV     #30,R0          ;;LOAD R0 WITH TEST NUMBER
1686  004506  012703  000001  1$:     MOV     #1,R3           ;[DEST] = +1
1687  004512  000257                  CCC                     ;SCOPE SYNC
1688
1689  004514  005303  2$:     DEC     R3              ;TEST THE DEC
1690
1691  004516  005703                  TST     R3              ;RESULT = 000000 ??
1692  004520  001402                  BEQ     TST31           ;;BR IF YES
1693
1694  004522  000000  3$:     HALT                            ;DEC DELIVERED THE WRONG RESULT
1695  004524  000770                  BR      1$              ;LOCK ON HARD ERROR
1696
1697                          ;;*****************************************************************
1698                          ;*TEST 31        BASIC "DEC  @#A" TEST
1699                          ;;*****************************************************************
1700  004526                  TST31:
1701  004526  012700  000031          MOV     #31,R0          ;;LOAD R0 WITH TEST NUMBER
1702  004532  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
1703  004536  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
1704  004542  005012  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
1705  004544  000257                  CCC                     ;SCOPE SYNC
1706
1707  004546  005337  063236  2$:     DEC     @#MBUF0         ;TEST THE DEC
1708
1709  004552  020412                  CMP     R4,(R2)         ;DID RESULT = 177777 ??
1710  004554  001402                  BEQ     TST32           ;;BR IF YES
1711
1712  004556  000000  3$:     HALT                            ;DEC DELIVERED WRONG RESULT
1713  004560  000770                  BR      1$              ;LOCK ON HARD ERROR
1714
1715                          ;;*****************************************************************
1716                          ;*TEST 32        BASIC "CLR X(R)" TESTS
1717                          ;;*****************************************************************
1718  004562                  TST32:
1719  004562  012700  000032          MOV     #32,R0          ;;LOAD R0 WITH TEST NUMBER
1720  004566  012702  063240          MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
1721  004572  005004                  CLR     R4              ;RESULT S / B = 000000
1722  004574  012705  063236  1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
1723  004600  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
```

```
1724  004604  000257                  CCC                     ;SCOPE SYNC
1725
1726  004606  005065  000002  2$:     CLR     2(R5)           ;TEST THE CLR
1727
1728  004612  020412                  CMP     R4,(R2)         ;RESULT = 0?
1729  004614  001402                  BEQ     TST33           ;;BR IF YES
1730
1731  004616  000000  3$:     HALT                            ;CLR FAILED TO ZERO [DEST]
1732  004620  000765                  BR      1$              ;LOCK ON HARD ERROR.
1733
1734                  ;;**************************************************************
1735                  ;*TEST 33       BASIC "ASL  RN" TEST WITH [DEST]=125252 AND C(0)
1736                  ;;**************************************************************
1737  004622          TST33:
1738  004622  012700  000033          MOV     #33,R0          ;;LOAD R0 WITH TEST NUMBER
1739  004626  012703  125252  1$:     MOV     #125252,R3      ;MAKE [DEST] = 125252
1740  004632  000257                  CCC                     ;MAKE C=0
1741
1742  004634  006303  2$:     ASL     R3                      ;TEST THE ASL - IT SHOULD SET "C"
1743
1744  004636  103402                  BCS     4$              ;BR IF "C" GOT SET
1745
1746  004640  000000  3$:     HALT                            ;ASL FAILED TO SET "C" BIT
1747  004642  000771                  BR      1$              ;LOCK ON HRD ERROR
1748
1749  004644  022703  052524  4$:     CMP     #52524,R3       ;WAS RESULT = 52524 ??
1750  004650  001402                  BEQ     TST34           ;;BR IF YES
1751
1752  004652  000000  5$:     HALT                            ;ASL DELIVERED THE WRONG RESULT
1753  004654  000764                  BR      1$              ;LOCK ON HARD ERROR
1754
1755                  ;;**************************************************************
1756                  ;*TEST 34       BASIC "ASL  RN" TEST WITH [DEST]=052525 AND C(1)
1757                  ;;**************************************************************
1758  004656          TST34:
1759  004656  012700  000034          MOV     #34,R0          ;;LOAD R0 WITH TEST NUMBER
1760  004662  012703  052525  1$:     MOV     #052525,R3      ;MAKE [DEST] = 052525
1761  004666  000261                  SEC                     ;MAKE C=1
1762
1763  004670  006303  2$:     ASL     R3                      ;TEST THE ASL - IT SHOULD CLR "C"
1764
1765  004672  103002                  BCC     4$              ;BR IF "C" GOT CLEARED
1766
1767  004674  000000  3$:     HALT                            ;ASL FAILED TO CLEAR "C"
1768  004676  000771                  BR      1$              ;LOCK ON HARD ERROR
1769
1770  004700  022703  125252  4$:     CMP     #125252,R3      ;RESULT = 125252 ??
1771  004704  001402                  BEQ     TST35           ;;BR IF YES
1772
1773  004706  000000  5$:     HALT                            ;ASL DELIVERED WRONG REULT
1774  004710  000764                  BR      1$              ;LOCK ON HARD ERROR
1775                  ;;**************************************************************
1776                  ;*TEST 35       BASIC "ROL  RN" TEST WITH [DEST]=125252 AND C(0)
1777                  ;;**************************************************************
1778  004712          TST35:
1779  004712  012700  000035          MOV     #35,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
1780  004716  012703  125252          1$:     MOV     #125252,R3      ;MAKE [DEST] = 125252
1781  004722  000257                          CCC                     ;MAKE C=0
1782
1783  004724  006103                  2$:     ROL     R3              ;TEST THE ROL - IT SHOULD SET C
1784
1785  004726  103402                          BCS     4$              ;BR IF "C" GOT SET
1786
1787  004730  000000                  3$:     HALT                    ;ROL FAILED TO SET "C"
1788  004732  000771                          BR      1$              ;LOCK ON HARD ERROR
1789
1790  004734  022703  052524          4$:     CMP     #052524,R3      ;RESULT = 052524 ??
1791  004740  001402                          BEQ     TST36           ;;BR IF YES
1792
1793  004742  000000                  5$:     HALT                    ;ROL DELIVERED WRONG RESULT
1794  004744  000764                          BR      1$              ;LOCK ON HARD ERROR
1795
1796                                  ;;**************************************************************
1797                                  ;*TEST 36       BASIC "ROL  RN" TEST WITH [DEST]=052524 AND C(1)
1798                                  ;;**************************************************************
1799  004746                          TST36:
1800  004746  012700  000036                  MOV     #36,R0          ;;LOAD R0 WITH TEST NUMBER
1801  004752  012703  052524          1$:     MOV     #052524,R3      ;MAKE [DEST] = 052524
1802  004756  000261                          SEC                     ;MAKE C=1
1803
1804  004760  006103                  2$:     ROL     R3              ;TEST THE ROL - IT SHOULD CLEAR C
1805
1806  004762  103002                          BCC     4$              ;BR IF "C" IS CLEAR
1807
1808  004764  000000                  3$:     HALT                    ;ROL FAILED TO CLEAR "C"
1809  004766  000771                          BR      1$              ;LOCK ON HARD ERROR
1810
1811  004770  022703  125251          4$:     CMP     #125251,R3      ;RESULT = 125251 ??
1812  004774  001402                          BEQ     TST37           ;;BR IF YES
1813
1814  004776  000000                  5$:     HALT                    ;ROL DELIVERED WRONG RESULT
1815  005000  000764                          BR      1$              ;LOCK ON HARD ERROR
1816
1817                                  ;;**************************************************************
1818                                  ;*TEST 37       BASIS "TSTB (R)" TEST - EVEN ADDRESS
1819                                  ;;**************************************************************
1820  005002                          TST37:
1821  005002  012700  000037                  MOV     #37,R0          ;;LOAD R0 WITH TEST NUMBER
1822  005006  012702  063254                  MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
1823  005012  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
1824  005016  000257                  1$:     CCC                     ;SCOPE SYNC
1825
1826  005020  105712                  2$:     TSTB    (R2)            ;TEST THE TSTB
1827
1828  005022  100402                          BMI     4$              ;BR IF "N" SET - IT SHOULD BE
1829
1830  005024  000000                  3$:     HALT                    ;TSTB FAILED TO SET "N"
1831  005026  000773                          BR      1$              ;LOCK ON HARD ERROR
1832
1833  005030  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
1834  005032  001403                          BEQ     TST40           ;;BR IF NOT
1835
```

```
1836  005034  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
1837  005036  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
1838  005040  000766                          BR      1$              ;LOCK ON HARD EROR
1839                                  ;;**********************************************************************
1840                                  ;*TEST 40        BASIS "TSTB (R)" TEST - ODD ADDRESS
1841                                  ;;**********************************************************************
1842  005042                          TST40:
1843  005042  012700  000040                 MOV     #40,R0          ;;LOAD R0 WITH TEST NUMBER
1844  005046  012702  063764                 MOV     #DWTB+6,R2      ;DEST ADDR = DWTB+6
1845  005052  012704  177401                 MOV     #177401,R4      ;RESULT S / B = 177401
1846  005056  012703  063765                 MOV     #DWTB+7,R3      ;DEST ADDR USED = DWTB+7
1847  005062  000257                  1$:     CCC                     ;SCOPE SYNC
1848
1849  005064  105713                  2$:     TSTB    (R3)            ;TEST THE TSTB
1850
1851  005066  100402                          BMI     4$              ;BR IF "N" SET - IT SHOULD BE
1852
1853  005070  000000                  3$:     HALT                    ;TSTB FAILED TO SET "N"
1854  005072  000773                          BR      1$              ;LOCK ON HARD ERROR
1855
1856  005074  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
1857  005076  001403                          BEQ     TST41           ;;BR IF NOT
1858
1859  005100  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
1860  005102  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
1861  005104  000766                          BR      1$              ;LOCK ON HARD EROR
1862
1863                                  ;;**********************************************************************
1864                                  ;*TEST 41        BASIC "TSTB @#A" TEST - EVEN ADDRESS
1865                                  ;;**********************************************************************
1866  005106                          TST41:
1867  005106  012700  000041                 MOV     #41,R0          ;;LOAD R0 WITH TEST NUMBER
1868  005112  012702  063252                 MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
1869  005116  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 177400
1870  005122  000257                  1$:     CCC                     ;SCOPE SYNC
1871
1872  005124  105737  063252          2$:     TSTB    @#DWTA+4        ;TEST THE TSTB
1873
1874  005130  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
1875
1876  005132  000000                  3$:     HALT                    ;TSTB FAILED TO SET "Z"
1877  005134  000772                          BR      1$              ;LOCK ON HARD ERROR
1878
1879  005136  020412                  4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
1880  005140  001403                          BEQ     TST42           ;;BR IF NOT
1881
1882  005142  000000                  5$:     HALT                    ;TSTB ALTERED [DEST]
1883  005144  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
1884  005146  000765                          BR      1$              ;LOCK ON HARD ERROR
1885
1886                                  ;;**********************************************************************
1887                                  ;*TEST 42        BASIC "TSTB @#A" TEST - ODD ADDRESS
1888                                  ;;**********************************************************************
1889  005150                          TST42:
1890  005150  012700  000042                 MOV     #42,R0          ;;LOAD R0 WITH TEST NUMBER
1891  005154  012702  063254                 MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
```

```
1892  005160  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
1893  005164  000257          1$:     CCC                     ;SCOPE SYNC
1894
1895  005166  105737  063255  2$:     TSTB    @#DWTA+7        ;TEST THE TSTB
1896
1897  005172  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
1898
1899  005174  000000          3$:     HALT                    ;TSTB FAILED TO SET "Z"
1900  005176  000772                  BR      1$              ;LOCK ON HARD ERROR
1901
1902  005200  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
1903  005202  001403                  BEQ     TST43           ;;BR IF NOT
1904
1905  005204  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
1906  005206  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
1907  005210  000765                  BR      1$              ;LOCK ON HARD ERROR
1908
1909                          ;;**********************************************************************
1910                          ;*TEST 43        BASIC "DECB 1(SP)"
1911                          ;;**********************************************************************
1912  005212                  TST43:
1913  005212  012700  000043          MOV     #43,R0          ;;LOAD R0 WITH TEST NUMBER
1914  005216  010605                  MOV     SP,R5           ;SAVE SP
1915  005220  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
1916  005224  010506          1$:     MOV     R5,SP
1917  005226  005046                  CLR     -(SP)           ;[DEST] = 000000
1918  005230  000257                  CCC                     ;SCOPE SYNC
1919
1920  005232  105366  000001  2$:     DECB    1(SP)           ;TEST THE DECB
1921
1922  005236  020416                  CMP     R4,(SP)         ;RESULT = 177400?
1923  005240  001402                  BEQ     4$              ;BR IF YES
1924
1925  005242  000000          3$:     HALT                    ;ERROR - DECB FAILED
1926  005244  000767                  BR      1$              ;LOCK ON HARD ERROR
1927
1928  005246  010506          4$:     MOV     R5,SP           ;RESET THE SP
1929
1930                          ;;**********************************************************************
1931                          ;*TEST 44        BASIC "MOV @#A,R" TEST
1932                          ;;**********************************************************************
1933  005250                  TST44:
1934  005250  012700  000044          MOV     #44,R0          ;;LOAD R0 WITH TEST NUMBER
1935  005254  005003          1$:     CLR     R3              ;[DEST] = 000000
1936  005256  000257                  CCC                     ;SCOPE SYNC
1937
1938  005260  013703  063222  2$:     MOV     @#ATA,R3        ;TEST THE MOV
1939
1940  005264  022703  063246          CMP     #DWTA,R3        ;RESULT = DWTA?
1941  005270  001402                  BEQ     TST45           ;;BR IF YES
1942
1943  005272  000000          3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
1944  005274  000767                  BR      1$              ;LOCK ON HARD ERROR
1945
1946                          ;;**********************************************************************
1947                          ;*TEST 45        BASIC "MOV #N,X(R)" TEST
```

```
1948                                    ;;*****************************************************************
1949  005276                            T$T45:
1950  005276  012700  000045                   MOV     #45,R0          ;;LOAD R0 WITH TEST NUMBER
1951  005302  012702  063240                   MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
1952  005306  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
1953  005312  012703  063236           1$:     MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
1954  005316  005012                           CLR     (R2)            ;[DEST] = 000000
1955  005320  000257                           CCC                     ;SCOPE SYNC
1956
1957  005322  012763  125252  000002   2$:     MOV     #125252,2(R3)   ;TEST THE MOV
1958
1959  005330  020412                           CMP     R4,(R2)         ;RESULT OK?
1960  005332  001402                           BEQ     T$T46           ;;BR IF YES
1961
1962  005334  000000                   3$:     HALT                    ;MOV DELIVERED WRONG RESULT
1963  005336  000765                           BR      1$              ;LOCK ON HARD ERROR
1964
1965                                    ;;*****************************************************************
1966                                    ;*TEST 46        BASIC "MOV #N,(R)" TEST
1967                                    ;;*****************************************************************
1968  005340                            T$T46:
1969  005340  012700  000046                   MOV     #46,R0          ;;LOAD R0 WITH TEST NUMBER
1970  005344  012703  063236                   MOV     #MBUF0,R3       ;DEST ADDR = MBUF0
1971  005350  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
1972  005354  005013           1$:     CLR     (R3)            ;[DEST] = 000000
1973  005356  000257                           CCC                     ;SCOPE SYNC
1974
1975  005360  012713  125252           2$:     MOV     #125252,(R3)    ;TEST THE MOV
1976
1977  005364  020413                           CMP     R4,(R3)         ;RESULT OK?
1978  005366  001402                           BEQ     T$T47           ;;BR IF YES
1979
1980  005370  000000                   3$:     HALT                    ;MOV DELIVERED WRONG RESULT
1981  005372  000770                           BR      1$              ;LOCK ON HARD ERROR
1982
1983                                    ;;*****************************************************************
1984                                    ;*TEST 47        BASIC "MOV (RA)+,RB" TEST
1985                                    ;;*****************************************************************
1986  005374                            T$T47:
1987  005374  012700  000047                   MOV     #47,R0          ;;LOAD R0 WITH TEST NUMBER
1988  005400  012705  063222           1$:     MOV     #ATA,R5         ;SRC ADDR = ATA
1989  005404  005003                           CLR     R3              ;[DEST] = 000000
1990  005406  000257                           CCC                     ;SCOPE SYNC
1991
1992  005410  012503                   2$:     MOV     (R5)+,R3        ;TEST THE MOV
1993
1994  005412  022703  063246                   CMP     #DWTA,R3        ;RESULT OK?
1995  005416  000402                           BR      4$              ;BR IF YES
1996
1997  005420  000000                   3$:     HALT                    ;MOV DELIVERED WRONG RESULT
1998  005422  000766                           BR      1$              ;LOCK ON HARD ERROR
1999
2000  005424  022705  063224           4$:     CMP     #ATA+2,R5       ;DID SRC REG GET INCREMENTED?
2001  005430  001402                           BEQ     TST50           ;;BR IF YES
2002
2003  005432  000000                   5$:     HALT                    ;MOV FAILED TO UPDATE SRC. REG.
```

```
2004  005434  000761                          BR      1$              ;LOCK ON HARD ERROR
2005
2006                                  ;;*****************************************************************
2007                                  ;*TEST 50        BASIC "MOV  @#A,@#B"
2008                                  ;;*****************************************************************
2009  005436                          TST50:
2010  005436  012700  000050                 MOV     #50,R0          ;;LOAD R0 WITH TEST NUMBER
2011  005442  012702  063242                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2012  005446  012704  063246                 MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2013  005452  005012                  1$:    CLR     (R2)            ;MAKE [DEST] = 000000
2014  005454  000257                          CCC                     ;SCOPE SYNC
2015
2016  005456  013737  063222  063242  2$:    MOV     @#ATA,@#MBUF1   ;TEST THE MOV
2017  005464  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ?
2018  005466  001402                          BEQ     TST1            ;;BR IF YES
2019
2020  005470  000000                  3$:    HALT                    ;MOV DELIVERED THE WRONG RESULT
2021  005472  000767                          BR      1$              ;LOCK ON HARD ERROR
2022
2023                                  ;;*****************************************************************
2024                                  ;*TEST 51        BASIC "MOV   X(R),PC" TEST
2025                                  ;;*****************************************************************
2026  005474                          TST51:
2027  005474  012700  000051                 MOV     #51,R0          ;;LOAD R0 WITH TEST NUMBER
2028  005500  012705  005506          1$:    MOV     #2$,R5          ;[R5] = 2$ (BASE ADDRESS)
2029  005504  000257                          CCC                     ;SCOPE SYNC
2030
2031  005506  016507  000010          2$:    MOV     4$-2$(R5),PC    ;TEST THE MOV - GO TO NEXT TEST VIA 4$
2032
2033  005512  000000                  3$:    HALT                    ;MOV FAILED TO LOAD THE PC
2034  005514  000771                          BR      1$              ;LOCK ON HARD ERROR
2035
2036  005516  005520                  4$:    .+2                     ;POINTER TO NEXT TEST
2037
2038                                  ;;*****************************************************************
2039                                  ;*TEST 52        BASIC "MOV  @#A,(R)" TEST
2040                                  ;;*****************************************************************
2041  005520                          TST52:
2042  005520  012700  000052                 MOV     #52,R0          ;;LOAD R0 WITH TEST NUMBER
2043  005524  012704  063246                 MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2044  005530  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2045  005534  005012                  1$:    CLR     (R2)            ;MAKE [DEST]=000000
2046  005536  000257                          CCC                     ;SCOPE SYNC - Z=0
2047
2048  005540  013712  063222          2$:    MOV     @#ATA,(R2)      ;TEST THE MOV
2049
2050  005544  020412                          CMP     R4,(R2)         ;DID RESULT = #DWTA ??
2051  005546  001402                          BEQ     TST3            ;;BR IF YES
2052
2053  005550  000000                  3$:    HALT                    ;MOV DELIVERED WRONG RESULT
2054  005552  000770                          BR      1$              ;LOCK ON HARD ERROR
2055
2056                                  ;;*****************************************************************
2057                                  ;*TEST 53        BASIC "MOV  X(RA),RB" TEST
2058                                  ;;*****************************************************************
2059  005554                          TST53:
```

```
2060  005554  012700  000053          MOV     #53,R0          ;;LOAD R0 WITH TEST NUMBER
2061  005560  012705  063222          MOV     #ATA,R5         ;[R5] = BASE ADDR FOR SOURCE (ATA)
2062  005564  005003          1$:     CLR     R3              ;MAKE [DEST] = 000000
2063  005566  000257                  CCC                     ;SCOPE SYNC
2064
2065  005570  016503  000004  2$:     MOV     4(R5),R3        ;TEST THE MOV
2066
2067  005574  022703  064554          CMP     #DBTA,R3        ;RESULT = #DBTA ??
2068  005600  001402                  BEQ     TST54           ;;BR IF YES
2069
2070  005602  000000          3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2071  005604  000767                  BR      1$              ;LOCK ON HARD ERROR
2072
2073                          ;;*****************************************************************
2074                          ;*TEST 54       BASIC "MOV  RA,-(RB)" TEST
2075                          ;;*****************************************************************
2076  005606                  TST54:
2077  005606  012700  000054          MOV     #54,R0          ;;LOAD R0 WITH TEST NUMBER
2078  005612  012702  063236          MOV     #MBUF0,R2       ;FINAL DEST ADDR = MBUF0
2079  005616  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2080  005622  012705  063240  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = TEMP2 + 2
2081  005626  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2082  005630  000257                  CCC                     ;SCOPE SYNC
2083
2084  005632  010445          2$:     MOV     R4,-(R5)        ;TEST THE MOV
2085
2086  005634  020412                  CMP     R4,(R2)         ;RESULT = 125252
2087  005636  001402                  BEQ     4$              ;BR IF YES
2088
2089  005640  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG ESULT
2090  005642  000767                  BR      1$              ;LOCK ON HARD ERROR
2091
2092  005644  020205          4$:     CMP     R2,R5           ;DID REGISTER GET DECREMENTED ?
2093  005646  001402                  BEQ     TST55           ;;BR IF YES
2094
2095  005650  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2096  005652  000763                  BR      1$              ;LOCK ON HARD ERROR
2097
2098                          ;;*****************************************************************
2099                          ;*TEST 55       BASIC "MOV   @#A,-(R)" TEST
2100                          ;;*****************************************************************
2101  005654                  TST55:
2102  005654  012700  000055          MOV     #55,R0          ;;LOAD R0 WITH TEST NUMBER
2103  005660  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2104  005664  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2105  005670  012705  063240  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = MBUF0+2
2106  005674  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2107  005676  000257                  CCC                     ;SCOPE SYNC
2108
2109  005700  013745  063222  2$:     MOV     @#ATA,-(R5)     ;TEST THE MOV
2110
2111  005704  020412                  CMP     R4,(R2)         ;RESULT = 000000
2112  005706  001402                  BEQ     4$              ;BR IF YES
2113
2114  005710  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2115  005712  000766                  BR      1$              ;LOCK ON HARD ERROR
```

```
2116
2117  005714  020502          4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED ??
2118  005716  001402                  BEQ     TST56           ;;BR IF YES
2119
2120  005720  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2121  005722  000762                  BR      1$              ;LOCK ON HARD ERROR
2122
2123                          ;;**************************************************************
2124                          ;*TEST 56       BASIC "MOV  (R),@#A" TEST
2125                          ;;**************************************************************
2126  005724                  TST56:
2127  005724  012700  000056          MOV     #56,R0          ;;LOAD R0 WITH TEST NUMBER
2128  005730  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2129  005734  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2130  005740  012705  063222          MOV     #ATA,R5         ;SOURCE ADDR = ATA
2131  005744  005012          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2132  005746  000257                  CCC                     ;SCOPE SYNC
2133
2134  005750  011537  063236  2$:     MOV     (R5),@#MBUF0    ;TEST THE MOV
2135
2136  005754  020412                  CMP     R4,(R2)         ;RESULT = #DWTA ??
2137  005756  001402                  BEQ     TST57           ;;BR IF YES
2138
2139  005760  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2140  005762  000770                  BR      1$              ;LOCK ON HARD ERROR
2141
2142                          ;;**************************************************************
2143                          ;*TEST 57       BASIC "MOV  -(R),@#A" TEST
2144                          ;;**************************************************************
2145  005764                  TST57:
2146  005764  012700  000057          MOV     #57,R0          ;;LOAD R0 WITH TEST NUMBER
2147  005770  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2148  005774  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2149  006000  012705  063224  1$:     MOV     #ATA+2,R5       ;INITIAL SOURCE ADDR = ATA+2
2150  006004  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2151  006006  000257                  CCC                     ;SCOPE SYNC
2152
2153  006010  014537  063236  2$:     MOV     -(R5),@#MBUF0   ;TEST THE MOV
2154
2155  006014  020412                  CMP     R4,(R2)         ;RESULT = #DWTA ?
2156  006016  001402                  BEQ     4$              ;BR IF YES
2157
2158  006020  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2159  006022  000766                  BR      1$              ;LOCK ON HARD ERROR
2160
2161  006024  022705  063222  4$:     CMP     #ATA,R5         ;DID THE SRC REG GET DECREMENTED ?
2162  006030  001402                  BEQ     TST60           ;;BR IF YES
2163
2164  006032  000000          5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REG
2165  006034  000761                  BR      1$              ;LOCK ON HARD ERROR
2166                          ;;**************************************************************
2167                          ;*TEST 60       BASIC "MOV   (RA),RB" TEST
2168                          ;;**************************************************************
2169  006036                  TST60:
2170  006036  012700  000060          MOV     #60,R0          ;;LOAD R0 WITH TEST NUMBER
2171  006042  012705  063222  1$:     MOV     #ATA,R5         ;INITIAL SOURCE ADDR = ATA
```

```
2172  006046  005003                    CLR     R3              ;MAKE [DEST] = 000000
2173  006050  000257                    CCC                     ;SCOPE SYNC
2174
2175  006052  012503            2$:     MOV     (R5)+,R3        ;TEST THE MOV
2176
2177  006054  022703  063246            CMP     #DWTA,R3        ;RESULT = #DWTA ?
2178  006060  001402                    BEQ     4$              ;BR IF YES
2179
2180  006062  000000            3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2181  006064  000766                    BR      1$              ;LOCK ON HARD ERROR
2182
2183  006066  022705  063224      4$:   CMP     #ATA+2,R5       ;DID SOURCE REG GET INCREMENTED
2184  006072  001402                    BEQ     TST61           ;;BR IF YES
2185
2186  006074  000000            5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REGISTER
2187  006076  000761                    BR      1$              ;LOCK ON HARD ERROR
2188
2189                            ;;**************************************************************
2190                            ;*TEST 61        BASIC "MOV   X(RA),RB" TEST
2191                            ;;**************************************************************
2192  006100                    TST61:
2193  006100  012700  000061            MOV     #61,R0          ;;LOAD R0 WITH TEST NUMBER
2194  006104  012705  063222            MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
2195  006110  005003            1$:     CLR     R3              ;MAKE [DEST] = 000000
2196  006112  000257                    CCC                     ;SCOPE SYNC
2197
2198  006114  016503  000002      2$:   MOV     2(R5),R3        ;TEST THE MOV
2199
2200  006120  022703  063756            CMP     #DWTB,R3        ;RESULT = #DWTB ?
2201  006124  001402                    BEQ     TST62           ;;BR IF YES
2202
2203  006126  000000            3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2204  006130  000767                    BR      1$              ;LOCK ON HARD ERROR
2205                            ;;**************************************************************
2206                            ;*TEST 62        BASIC "MOV   @X(RA),RB" TEST
2207                            ;;**************************************************************
2208  006132                    TST62:
2209  006132  012700  000062            MOV     #62,R0          ;;LOAD R0 WITH TEST NUMBER
2210  006136  012737  063250  063240     MOV     #DWTA+2,@#MBUF0+2 ;SET UP ADDRESS TABLE MBUF0
2211  006144  012705  063236            MOV     #MBUF0,R5       ;BASE ADDRESS IN R5
2212  006150  005003            1$:     CLR     R3              ;MAKE [DEST] = 000000
2213  006152  000257                    CCC                     ;SCOPE SYNC
2214
2215  006154  017503  000002      2$:   MOV     @2(R5),R3       ;TEST THE MOV
2216
2217  006160  022703  177777            CMP     #-1,R3          ;RESULT = 177777
2218  006164  001402                    BEQ     TST63           ;;BR IF YES
2219
2220  006166  000000            3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2221  006170  000767                    BR      1$              ;LOCK ON HARD ERROR
2222
2223                            ;;**************************************************************
2224                            ;*TEST 63        BASIC "MOV  (R)+,X(R)" TEST
2225                            ;;**************************************************************
2226  006172                    TST63:
2227  006172  012700  000063            MOV     #63,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2228  006176  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2229  006202  012702  063244          MOV     #MBUF1+2,R2     ;FINAL DEST ADDR = MBUF1+2
2230  006206  010437  063236          MOV     R4,@#MBUF0      ;SOURCE OPERAND = 125252
2231  006212  012705  063236  1$:     MOV     #MBUF0,R5       ;[R5] = INITIAL SRC ADDR = MBUF0
2232  006216  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2233  006220  000257                  CCC                     ;SCOPE SYNC
2234
2235  006222  012565  000004  2$:     MOV     (R5)+,4(R5)     ;TEST THE MOV
2236
2237  006226  020412                  CMP     R4,(R2)         ;RESULT = 125252 ?
2238  006230  001402                  BEQ     4$              ;BR IF YES
2239
2240  006232  000000  3$:     HALT                            ;MOV    DELIVERED WRONG RESULT
2241  006234  000766                  BR      1$              ;LOCK ON HARD ERROR
2242
2243  006236  022705  063240  4$:     CMP     #MBUF0+2,R5     ;DID REGISTER GET INCREMENTED ?
2244  006242  001402                  BEQ     TST64           ;;BR IF YES
2245
2246  006244  000000  5$:     HALT                            ;MOV FAILED TO UPDATE REGISTER
2247  006246  000761                  BR      1$              ;LOCK ON HARD ERROR
2248
2249                          ;;**************************************************************
2250                          ;*TEST 64       BASIC "CMP  R,@#A" TEST WITH [R] = [A]
2251                          ;;**************************************************************
2252  006250                  TST64:
2253  006250  012700  000064          MOV     #64,R0          ;;LOAD R0 WITH TEST NUMBER
2254                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT1
2255  006254  032737  000002  063160  BIT     #BIT1,@#BPTLOC  ;BREAKPOINT HALT SET ??
2256  006262  001401                  BEQ     .+4             ;BR IF NOT
2257  006264  000000                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2258  006266  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2259  006272  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2260  006276  010405  1$:     MOV     R4,R5           ;[R5] = SOURCE OP = 125252
2261  006300  010412                  MOV     R4,(R2)         ;MAKE [DEST] = 125252
2262  006302  000257                  CCC                     ;SCOPE SYNC
2263
2264  006304  020537  063236  2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2265
2266  006310  001402                  BEQ     4$              ;BR IF "Z" WAS SET - IT SHOULD BE
2267
2268  006312  000000  3$:     HALT                            ;CMP FAILED TO SET "Z"
2269  006314  000770                  BR      1$              ;LOCK ON HARD ERROR
2270
2271  006316  020412  4$:     CMP     R4,(R2)         ;IS RESULT STILL = 125252 ?
2272  006320  001402                  BEQ     TST65           ;;BR IF YES
2273
2274  006322  000000  5$:     HALT                            ;CMP ALTERED [DEST]
2275  006324  000764                  BR      1$              ;LOCK ON HARD ERROR
2276
2277                          ;;**************************************************************
2278                          ;*TEST 65       BASIC "CMP  R,@#A" WITH [R] NOT EQUAL TO [A]
2279                          ;;**************************************************************
2280  006326                  TST65:
2281  006326  012700  000065          MOV     #65,R0          ;;LOAD R0 WITH TEST NUMBER
2282  006332  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2283  006336  012704  125252          MOV     #125252,R4      ;MAKE RESULT S / B = 125252
```

```
2284  006342  005005                    1$:     CLR     R5              ;[R5] = SOURCE OP = 000000
2285  006344  010412                            MOV     R4,(R2)         ;MAKE [DEST] = 125252
2286  006346  000277                            SCC                     ;SCOPE SYNC - MAKE Z=1
2287
2288  006350  020537  063236            2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2289
2290  006354  001002                            BNE     TST66           ;;BR IF Z=0 - IT SHOULD BE
2291
2292  006356  000000                    3$:     HALT                    ;CMP FAILED TO CLEAR "Z"
2293  006360  000770                            BR      1$              ;LOCK ON HARD ERROR
2294
2295                                    ;;**************************************************************
2296                                    ;*TEST 66       BASIC "BIS #N,@#A" TEST - N=177777,[A]=000000
2297                                    ;;**************************************************************
2298  006362                            TST66:
2299  006362  012700  000066                    MOV     #66,R0          ;;LOAD R0 WITH TEST NUMBER
2300  006366  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2301  006372  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
2302  006376  005012                    1$:     CLR     (R2)            ;[DEST] = 000000
2303  006400  000257                            CCC                     ;SCOPE SYNC
2304
2305  006402  052737  177777  063236    2$:     BIS     #-1,@#MBUF0     ;TEST THE BIS
2306
2307  006410  020412                            CMP     R4,(R2)         ;RESULT OK?
2308  006412  001402                            BEQ     TST67           ;;BR IF YES
2309
2310  006414  000000                    3$:     HALT                    ;BIS FAILED TO SET ALL BITS IN BITFLG
2311  006416  000767                            BR      1$              ;LOCK ON HARD ERROR
2312
2313                                    ;;**************************************************************
2314                                    ;*TEST 67       BASIC "BIC  #N,@#A" TEST
2315                                    ;;**************************************************************
2316  006420                            TST67:
2317  006420  012700  000067                    MOV     #67,R0          ;;LOAD R0 WITH TEST NUMBER
2318  006424  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2319  006430  012704  000077                    MOV     #77,R4          ;RESULT S / B = 77
2320  006434  012712  177777            1$:     MOV     #-1,(R2)        ;MAKE [DEST] = 177777
2321  006440  000257                            CCC                     ;SCOPE SYNC
2322
2323  006442  042737  177700  063236    2$:     BIC     #177700,@#MBUF0 ;TEST THE BIC
2324
2325  006450  020412                            CMP     R4,(R2)         ;DID RESULT = 77 ?
2326  006452  001402                            BEQ     TST70           ;;BR IF YES
2327
2328  006454  000000                    3$:     HALT                    ;BIC DELIVERED THE WRONG RESULT
2329  006456  000766                            BR      1$              ;LOCK ON HARD ERROR
2330
2331                                    ;;**************************************************************
2332                                    ;*TEST 70       BASIC "BIC #N,R" TEST
2333                                    ;;**************************************************************
2334  006460                            TST70:
2335  006460  012700  000070                    MOV     #70,R0          ;;LOAD R0 WITH TEST NUMBER
2336  006464  005003                    1$:     CLR     R3              ;[DEST] = 177777
2337  006466  005103                            COM     R3
2338  006470  000257                            CCC                     ;SCOPE SYNC
2339
```

```
2340  006472  042703  177400          2$:     BIC     #177400,R3      ;TEST THE BIC
2341
2342  006476  022703  000377                  CMP     #377,R3         ;RESULT OK?
2343  006502  001402                          BEQ     TST71           ;;BR IF YES
2344
2345  006504  000000                  3$:     HALT                    ;BIC FAILED TO CLEAR HI-BYTE
2346  006506  000766                          BR      1$              ;LOCK ON HARD ERROR
2347
2348                                  ;;**********************************************************************
2349                                  ;*TEST 71        BASIC "BIC #N,2(SP)" TEST
2350                                  ;;**********************************************************************
2351  006510                          TST71:
2352  006510  012700  000071                  MOV     #71,R0          ;;LOAD R0 WITH TEST NUMBER
2353  006514  012704  000357                  MOV     #357,R4         ;RESULT S / B = 357
2354  006520  010605                          MOV     SP,R5           ;SAVE SP
2355  006522  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
2356  006524  012746  000377                  MOV     #377,-(SP)      ;[DEST] = 377 PUT ON STACK
2357  006530  005746                          TST     -(SP)           ;DECREMENT SP
2358  006532  000257                          CCC                     ;SCOPE SYNC
2359
2360  006534  042766  000020  000002  2$:     BIC     #20,2(SP)       ;TEST THE BIC - CLEAR BIT 4
2361
2362  006542  010602                          MOV     SP,R2           ;[R2] = DEST ADDR
2363  006544  005722                          TST     (R2)+
2364  006546  020412                          CMP     R4,(R2)         ;RESULT = 357?
2365  006550  001402                          BEQ     4$              ;BR IF YES
2366
2367  006552  000000                  3$:     HALT                    ;BIC FAILED TO CLR BIT2 OF DEST
2368  006554  000762                          BR      1$              ;LOCK ON HARD ERROR
2369
2370  006556  010506                  4$:     MOV     R5,SP
2371
2372                                  ;;**********************************************************************
2373                                  ;*TEST 72        BASIC "ADD  #N,RN " TEST
2374                                  ;;**********************************************************************
2375  006560                          TST72:
2376  006560  012700  000072                  MOV     #72,R0          ;;LOAD R0 WITH TEST NUMBER
2377  006564  012703  000002          1$:     MOV     #2,R3           ;MAKE [DEST] = 2
2378  006570  000257                          CCC                     ;SCOPE SYNC
2379
2380  006572  062703  000002          2$:     ADD     #2,R3           ;TEST THE ADD
2381
2382  006576  022703  000004                  CMP     #4,R3           ;RESULT = 4 ?
2383  006602  001402                          BEQ     TST73           ;;BR IF YES
2384
2385  006604  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
2386  006606  000766                          BR      1$              ;LOCK ON HARD ERROR
2387
2388                                  ;;**********************************************************************
2389                                  ;*TEST 73        BASIC "ADD  #N,(R) " TEST
2390                                  ;;**********************************************************************
2391  006610                          TST73:
2392  006610  012700  000073                  MOV     #73,R0          ;;LOAD R0 WITH TEST NUMBER
2393  006614  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2394  006620  012704  000004                  MOV     #4,R4           ;RESULT S / B = 4
2395  006624  012712  000002          1$:     MOV     #2,(R2)         ;MAKE [DEST] = 2
```

```
2396  006630  000257                          CCC                     ;SCOPE SYNC
2397
2398  006632  062712  000002          2$:     ADD     #2,(R2) ;TEST THE ADD
2399
2400  006636  020412                          CMP     R4,(R2)         ;RESULT = 4 ?
2401  006640  001402                          BEQ     TST74           ;;BR IF YES
2402
2403  006642  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
2404  006644  000767                          BR      1$              ;LOCK ON HARD ERROR
2405
2406                                  ;;**********************************************************************
2407                                  ;*TEST 74       BASIC "ADD  #N,X(R)" TEST
2408                                  ;;**********************************************************************
2409  006646                          TST74:
2410  006646  012700  000074                  MOV     #74,R0          ;;LOAD R0 WITH TEST NUMBER
2411  006652  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
2412  006656  012702  063240                  MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0 + 2
2413  006662  012705  063236          1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2414  006666  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
2415  006670  000257                          CCC                     ;SCOPE SYNC
2416
2417  006672  062765  000002  000002  2$:     ADD     #2,2(R5)        ;TEST THE ADD
2418
2419  006700  020412                          CMP     R4,(R2)         ;RESULT = 2 ?
2420  006702  001402                          BEQ     TST75           ;;BR IF YES
2421
2422  006704  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
2423  006706  000765                          BR      1$              ;LOOP ON HARD ERROR
2424
2425                                  ;;**********************************************************************
2426                                  ;*TEST 75       BASIC "CMPB #N,(SP)+" TEST
2427                                  ;;**********************************************************************
2428  006710                          TST75:
2429  006710  012700  000075                  MOV     #75,R0          ;;LOAD R0 WITH TEST NUMBER
2430  006714  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
2431  006720  010605                          MOV     SP,R5           ;SAVE SP
2432  006722  010602                          MOV     SP,R2           ;SET UP DEST ADDR
2433  006724  005742                          TST     -(R2)           ;R2 CONTAINS DEST ADDR
2434  006726  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
2435  006730  010446                          MOV     R4,-(SP)        ;MAKE [DEST] = 177400
2436  006732  000257                          CCC                     ;SCOPE SYNC - "Z" = 0
2437
2438  006734  122726  000000          2$:     CMPB    #0,(SP)+        ;TEST THE CMPB
2439
2440  006740  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2441
2442  006742  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
2443  006744  000770                          BR      1$              ;LOCK ON HARD ERROR
2444
2445  006746  020506          4$:     CMP     R5,SP           ;DID SP GET UPDATED BY 2?
2446  006750  001402                          BEQ     6$              ;BR IF YES
2447
2448  006752  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE SP PROPERLY
2449  006754  000764                          BR      1$              ;LOCK ON HARD ERROR
2450
2451  006756  020412                  6$:     CMP     R4,(R2)         ;[DEST] ALTERED?
```

```
2452  006760  001402                          BEQ     TST76           ;;BR IF NOT
2453
2454  006762  000000                  7$:     HALT                    ;CMPB MODIFIED [DEST]
2455  006764  000760                          BR      1$              ;LOCK ON HARD ERROR
2456
2457                                  ;;**********************************************************************
2458                                  ;*TEST 76       BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST EVEN
2459                                  ;;**********************************************************************
2460  006766                          TST76:
2461  006766  012700  000076                 MOV     #76,R0          ;;LOAD R0 WITH TEST NUMBER
2462  006772  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
2463  006776  012702  063250                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
2464  007002  012705  063254          1$:    MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
2465  007006  010203                         MOV     R2,R3           ;R3 GETS DEST ADDR
2466  007010  000257                         CCC                     ;SCOPE SYNC
2467
2468  007012  122523                  2$:    CMPB    (R5)+,(R3)+     ;TEST THE CMPB
2469
2470  007014  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
2471
2472  007016  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
2473  007020  000770                          BR      1$              ;LOCK ON HARD ERROR
2474
2475  007022  022703  063251          4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
2476  007026  001402                          BEQ     6$              ;BR IF YES
2477
2478  007030  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
2479  007032  000763                          BR      1$              ;LOCK ON HARD ERROR
2480
2481  007034  022705  063255          6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
2482  007040  001402                          BEQ     8$              ;BR IF YES
2483
2484  007042  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
2485  007044  000756                          BR      1$              ;LOCK ON HARD ERROR
2486
2487  007046  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
2488  007050  001403                          BEQ     TST77           ;;BR IF NOT
2489
2490  007052  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
2491  007054  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2492  007056  000751                          BR      1$              ;LOCK ON HARD ERROR
2493
2494                                  ;;**********************************************************************
2495                                  ;*TEST 77       BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST ODD
2496                                  ;;**********************************************************************
2497  007060                          TST77:
2498  007060  012700  000077                 MOV     #77,R0          ;;LOAD R0 WITH TEST NUMBER
2499  007064  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
2500  007070  012702  063250                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
2501  007074  012705  063253          1$:    MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
2502  007100  012703  063251                 MOV     #DWTA+3,R3      ;R3 GETS DEST ADDR+1
2503  007104  000257                         CCC                     ;SCOPE SYNC
2504
2505  007106  122523                  2$:    CMPB    (R5)+,(R3)+     ;TEST THE CMPB
2506
2507  007110  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
```

```
2508
2509  007112  000000            3$:   HALT                    ;CMPB FAILED TO SET "Z"
2510  007114  000767                  BR      1$              ;LOCK ON HARD ERROR
2511
2512  007116  022703  063252    4$:   CMP     #DWTA+4,R3      ;DID DEST REG GET UPDATED?
2513  007122  001402                  BEQ     6$              ;BR IF YES
2514
2515  007124  000000            5$:   HALT                    ;CMPB FAILED TO UPDATE DEST REG
2516  007126  000762                  BR      1$              ;LOCK ON HARD ERROR
2517
2518  007130  022705  063254    6$:   CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
2519  007134  001402                  BEQ     8$              ;BR IF YES
2520
2521  007136  000000            7$:   HALT                    ;CMPB FAILED TO UPDATE SRC REG
2522  007140  000755                  BR      1$              ;LOCK ON HARD ERROR
2523
2524  007142  020412            8$:   CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
2525  007144  001403                  BEQ     TST100          ;;BR IF NOT
2526
2527  007146  000000            9$:   HALT                    ;CMPB DELIVERED A RESULT
2528  007150  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2529  007152  000750                  BR      1$              ;LOCK ON HARD ERROR
2530
2531                        ;;*****************************************************************
2532                        ;*TEST 100      BASIC "CMPB (RA)+,(RB)+" - SRC / EVEN,DEST / ODD
2533                        ;;*****************************************************************
2534  007154            TST100:
2535  007154  012700  000100          MOV     #100,R0         ;;LOAD R0 WITH TEST NUMBER
2536  007160  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
2537  007164  012702  063252          MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
2538  007170  012705  063254    1$:   MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
2539  007174  012703  063253          MOV     #DWTA+5,R3      ;R3 GETS DEST ADDR
2540  007200  000257                  CCC                     ;SCOPE SYNC
2541
2542  007202  122523            2$:   CMPB    (R5)+,(R3)+     ;TEST THE CMPB
2543
2544  007204  001402                  BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
2545
2546  007206  000000            3$:   HALT                    ;CMPB FAILED TO SET "Z"
2547  007210  000767                  BR      1$              ;LOCK ON HARD ERROR
2548
2549  007212  022703  063254    4$:   CMP     #DWTA+6,R3      ;DID DEST REG GET UPDATED?
2550  007216  001402                  BEQ     6$              ;BR IF YES
2551
2552  007220  000000            5$:   HALT                    ;CMPB FAILED TO UPDATE DEST REG
2553  007222  000762                  BR      1$              ;LOCK ON HARD ERROR
2554
2555  007224  022705  063255    6$:   CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
2556  007230  001402                  BEQ     8$              ;BR IF YES
2557
2558  007232  000000            7$:   HALT                    ;CMPB FAILED TO UPDATE SRC REG
2559  007234  000755                  BR      1$              ;LOCK ON HARD ERROR
2560
2561  007236  020412            8$:   CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
2562  007240  001403                  BEQ     TST101          ;;BR IF NOT
2563
```

```
2564  007242  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
2565  007244  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2566  007246  000750                          BR      1$              ;LOCK ON HARD ERROR
2567
2568                                  ;;**********************************************************************
2569                                  ;*TEST 101       BASIC "CMPB (RA)+,(RB)+" - SRC / ODD,DEST / EVEN
2570                                  ;;**********************************************************************
2571  007250                          TST101:
2572  007250  012700  000101                  MOV     #101,R0         ;;LOAD R0 WITH TEST NUMBER
2573  007254  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
2574  007260  012702  063250                  MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
2575  007264  012705  063253          1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
2576  007270  010203                          MOV     R2,R3           ;R3 GETS DEST ADDR
2577  007272  000257                          CCC                     ;SCOPE SYNC
2578
2579  007274  122523                  2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
2580
2581  007276  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
2582
2583  007300  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
2584  007302  000770                          BR      1$              ;LOCK ON HARD ERROR
2585
2586  007304  022703  063251          4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
2587  007310  001402                          BEQ     6$              ;BR IF YES
2588
2589  007312  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
2590  007314  000763                          BR      1$              ;LOCK ON HARD ERROR
2591
2592  007316  022705  063254          6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
2593  007322  001402                          BEQ     8$              ;BR IF YES
2594
2595  007324  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
2596  007326  000756                          BR      1$              ;LOCK ON HARD ERROR
2597
2598  007330  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
2599  007332  001403                          BEQ     TST102          ;;BR IF NOT
2600
2601  007334  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
2602  007336  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
2603  007340  000751                          BR      1$              ;LOCK ON HARD ERROR
2604
2605                                  ;;**********************************************************************
2606                                  ;*TEST 102       BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN
2607                                  ;;**********************************************************************
2608  007342                          TST102:
2609  007342  012700  000102                  MOV     #102,R0         ;;LOAD R0 WITH TEST NUMBER
2610  007346  012702  063242                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2611  007352  012703  063236                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
2612  007356  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
2613  007362  012705  064554          1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
2614  007366  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
2615  007372  000257                          CCC                     ;SCOPE SYNC
2616
2617  007374  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
2618
2619  007400  020412                          CMP     R4,(R2)         ;RESULT OK?
```

```
MAINDEC-11-DQKDA-A   KD11-K BASIC LOGIC TESTS   MACY11 27(1006)  08-FEB-77  16:23  PAGE 50
DQKDAA.P11    08-FEB-77 16:17                   T102    BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN

 2620  007402  001402                          BEQ     4$              ;BR IF YES
 2621
 2622  007404  000000C                 3$:     HALT                    ;MOV DELIVERED WRONG RESULT
 2623  007406  000765                          BR      1$              ;LOCK ON HARD ERROR
 2624
 2625  007410  022705  064555          4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
 2626  007414  001402                          BEQ     TST103          ;;BR IF YES
 2627
 2628  007416  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
 2629  007420  000760                          BR      1$              ;LOCK ON HARD ERROR
 2630                                  ;;**********************************************************************
 2631                                  ;*TEST 103       BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST ODD
 2632                                  ;;**********************************************************************
 2633  007422                          TST103:
 2634  007422  012700  000103                  MOV     #103,R0         ;;LOAD R0 WITH TEST NUMBER
 2635  007426  012702  063242                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 2636  007432  012703  063236                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 2637  007436  012704  000777                  MOV     #777,R4         ;RESULT S / B = 777
 2638  007442  012705  064561          1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
 2639  007446  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
 2640  007452  000257                          CCC                     ;SCOPE SYNC
 2641
 2642  007454  112563  000005          2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
 2643
 2644  007460  020412                          CMP     R4,(R2)         ;RESULT OK?
 2645  007462  001402                          BEQ     4$              ;BR IF YES
 2646
 2647  007464  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
 2648  007466  000765                          BR      1$              ;LOCK ON HARD ERROR
 2649
 2650  007470  022705  064562          4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
 2651  007474  001402                          BEQ     TST104          ;;BR IF YES
 2652
 2653  007476  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
 2654  007500  000760                          BR      1$              ;LOCK ON HARD ERROR
 2655                                  ;;**********************************************************************
 2656                                  ;*TEST 104       BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST ODD
 2657                                  ;;**********************************************************************
 2658  007502                          TST104:
 2659  007502  012700  000104                  MOV     #104,R0         ;;LOAD R0 WITH TEST NUMBER
 2660  007506  012702  063242                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 2661  007512  012703  063236                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 2662  007516  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 2663  007522  012705  064554          1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
 2664  007526  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
 2665  007532  000257                          CCC                     ;SCOPE SYNC
 2666
 2667  007534  112563  000005          2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
 2668
 2669  007540  020412                          CMP     R4,(R2)         ;RESULT OK?
 2670  007542  001402                          BEQ     4$              ;BR IF YES
 2671
 2672  007544  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
 2673  007546  000765                          BR      1$              ;LOCK ON HARD ERROR
 2674
 2675  007550  022705  064555          4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
```

```
2676  007554  001402                        BEQ     TST105          ;;BR IF YES
2677
2678  007556  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
2679  007560  000760                        BR      1$              ;LOCK ON HARD ERROR
2680                                ;;**************************************************************
2681                                ;*TEST 105      BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST EVEN
2682                                ;;**************************************************************
2683  007562                        TST105:
2684  007562  012700  000105               MOV     #105,R0         ;;LOAD R0 WITH TEST NUMBER
2685  007566  012702  063242               MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
2686  007572  012703  063236               MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
2687  007576  012704  177401               MOV     #177401,R4      ;RESULT S / B = 177401
2688  007602  012705  064561        1$:    MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
2689  007606  012712  177777               MOV     #-1,(R2)        ;[DEST] = 177777
2690  007612  000257                        CCC                     ;SCOPE SYNC
2691
2692  007614  112563  000004        2$:    MOVB    (R5)+,4(R3)     ;TEST THE MOVB
2693
2694  007620  020412                        CMP     R4,(R2)         ;RESULT OK?
2695  007622  001402                        BEQ     4$              ;BR IF YES
2696
2697  007624  000000                3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2698  007626  000765                        BR      1$              ;LOCK ON HARD ERROR
2699
2700  007630  022705  064562        4$:    CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
2701  007634  001402                        BEQ     TST106          ;;BR IF YES
2702
2703  007636  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
2704  007640  000760                        BR      1$              ;LOCK ON HARD ERROR
2705
2706                                ;;**************************************************************
2707                                ;*TEST 106      BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST EVEN
2708                                ;;**************************************************************
2709  007642                        TST106:
2710  007642  012700  000106               MOV     #106,R0         ;;LOAD R0 WITH TEST NUMBER
2711  007646  012702  063236               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2712  007652  012704  177401               MOV     #177401,R4      ;RESULT S / B = 177401
2713  007656  012705  063756               MOV     #DWTB,R5        ;SRC ADDR = DWTB
2714  007662  010203                1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
2715  007664  012713  177777               MOV     #-1,(R3)        ;[DEST] = 177400
2716  007670  000257                        CCC                     ;SCOPE SYNC
2717
2718  007672  116523  000002        2$:    MOVB    2(R5),(R3)+     ;TEST THE MOVB
2719
2720  007676  020412                        CMP     R4,(R2)         ;RESULT OK?
2721  007700  001402                        BEQ     4$              ;BR IF YES
2722
2723  007702  000000                3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2724  007704  000766                        BR      1$              ;LOCK ON HARD ERROR
2725
2726  007706  022703  063237        4$:    CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
2727  007712  001402                        BEQ     TST107          ;;BR IF YES
2728
2729  007714  000000                5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
2730  007716  000761                        BR      1$              ;LOCK ON HARD ERROR
2731
```

```
                                ;;*****************************************************************
                                ;*TEST 107      BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST EVEN
                                ;;*****************************************************************
007720                          TST107:
007720  012700  000107                MOV     #107,R0         ;;LOAD R0 WITH TEST NUMBER
007724  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
007730  012704  177401                MOV     #177401,R4      ;RESULT S / B = 177401
007734  012705  064560                MOV     #DBTB,R5        ;SRC ADDR = DBTB
007740  010203          1$:           MOV     R2,R3           ;R3 GETS DEST ADDR
007742  012713  177777                MOV     #-1,(R3)        ;[DEST] = 177777
007746  000257                      CCC                     ;SCOPE SYNC
007750  116523  000001        2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
007754  020412                      CMP     R4,(R2)         ;RESULT OK?
007756  001402                      BEQ     4$              ;BR IF YES
007760  000000          3$:         HALT                    ;MOVB DELIVERED WRONG RESULT
007762  000766                      BR      1$              ;LOCK ON HARD ERROR
007764  022703  063237    4$:       CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
007770  001402                      BEQ     TST110          ;;BR IF YES
007772  000000          5$:         HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
007774  000761                      BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 110      BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST ODD
                                ;;*****************************************************************
007776                          TST110:
007776  012700  000110                MOV     #110,R0         ;;LOAD R0 WITH TEST NUMBER
010002  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
010006  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
010012  012705  063756                MOV     #DWTB,R5        ;SRC ADDR = DWTB
010016  012703  063237          1$:   MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR
010022  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
010026  000257                      CCC                     ;SCOPE SYNC
010030  116523  000002        2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
010034  020412                      CMP     R4,(R2)         ;RESULT OK?
010036  001402                      BEQ     4$              ;BR IF YES
010040  000000          3$:         HALT                    ;MOVB DELIVERED WRONG RESULT
010042  000765                      BR      1$              ;LOCK ON HARD ERROR
010044  022703  063240    4$:       CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
010050  001402                      BEQ     TST111          ;;BR IF YES
010052  000000          5$:         HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
010054  000760                      BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 111      BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST ODD
                                ;;*****************************************************************
010056                          TST111:
```

```
2788  010056  012700  000111            MOV     #111,R0         ;;LOAD R0 WITH TEST NUMBER
2789  010062  012702  063236            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2790  010066  012704  000777            MOV     #777,R4         ;RESULT S / B = 777
2791  010072  012705  064560            MOV     #DBTB,R5        ;SRC ADDR = DBTB
2792  010076  012703  063237    1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR = MBUF0+1
2793  010102  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
2794  010106  000257                    CCC                     ;SCOPE SYNC
2795
2796  010110  116523  000001    2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
2797
2798  010114  020412                    CMP     R4,(R2)         ;RESULT OK?
2799  010116  001402                    BEQ     4$              ;BR IF YES
2800
2801  010120  000000    3$:     HALT                            ;MOVB DELIVERED WRONG RESULT
2802  010122  000765                    BR      1$              ;LOCK ON HARD ERROR
2803
2804  010124  022703  063240    4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
2805  010130  001402                    BEQ     TST112          ;;BR IF YES
2806
2807  010132  000000    5$:     HALT                            ;MOVB FAILED TO AUTO INCREMENT DEST REG
2808  010134  000760                    BR      1$              ;LOCK ON HARD ERROR
2809
2810                      ;;*****************************************************************
2811                      ;*TEST 112      BASIC "MOVB -(RA),RB" TEST - SRC EVEN ADDR
2812                      ;;*****************************************************************
2813  010136              TST112:
2814  010136  012700  000112            MOV     #112,R0         ;;LOAD R0 WITH TEST NUMBER
2815  010142  012705  063255    1$:     MOV     #DWTA+7,R5      ;SRC ADDR = DWTA+7
2816  010146  005003                    CLR     R3              ;[DEST] = 000000
2817  010150  000257                    CCC                     ;SCOPE SYNC
2818
2819  010152  114503    2$:     MOVB    -(R5),R3        ;TEST THE MOVB
2820
2821  010154  022703  177777            CMP     #-1,R3          ;RESULT OK?
2822  010160  001402                    BEQ     4$              ;BR IF YES
2823
2824  010162  000000    3$:     HALT                            ;MOVB FAILED - WRONG RESULT
2825  010164  000766                    BR      1$              ;LOCK ON HARD ERROR
2826
2827  010166  022705  063254    4$:     CMP     #DWTA+6,R5      ;SRC REG GET DECREMENTED?
2828  010172  001402                    BEQ     TST113          ;;BR IF YES
2829
2830  010174  000000    5$:     HALT                            ;MOVB FAILED TO UPDATE SRC REG
2831  010176  000761                    BR      1$              ;LOCK ON HARD ERROR
2832                      ;;*****************************************************************
2833                      ;*TEST 113      BASIC "MOVB -(RA),RB" TEST - SRC ODD ADDR
2834                      ;;*****************************************************************
2835  010200              TST113:
2836  010200  012700  000113            MOV     #113,R0         ;;LOAD R0 WITH TEST NUMBER
2837  010204  012705  063254    1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
2838  010210  005003                    CLR     R3              ;[DEST] = 000000
2839  010212  000257                    CCC                     ;SCOPE SYNC
2840
2841  010214  114503    2$:     MOVB    -(R5),R3        ;TEST THE MOVB
2842
2843  010216  022703  177777            CMP     #-1,R3          ;RESULT OK?
```

```
2844  010222  001402                    BEQ     4$              ;BR IF YES
2845
2846  010224  000000              3$:   HALT                    ;MOVB FAILED - WRONG RESULT
2847  010226  000766                    BR      1$              ;LOCK ON HARD ERROR
2848
2849  010230  022705  063253      4$:   CMP     #DWTA+5,R5      ;SRC REG GET DECREMENTED?
2850  010234  001402                    BEQ     TST114          ;;BR IF YES
2851
2852  010236  000000              5$:   HALT                    ;MOVB FAILED TO UPDATE SRC REG
2853  010240  000761                    BR      1$              ;LOCK ON HARD ERROR
2854
2855                              ;;*****************************************************************
2856                              ;*TEST 114      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR EVEN
2857                              ;;*****************************************************************
2858  010242                      TST114:
2859  010242  012700  000114            MOV     #114,R0         ;;LOAD R0 WITH TEST NUMBER
2860  010246  010605                    MOV     SP,R5           ;SAVE SP
2861  010250  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
2862  010254  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
2863  010256  012703  064554            MOV     #DBTA,R3        ;SRC ADDR = DBTA
2864  010262  012746  177777            MOV     #-1,-(SP)       ;[DEST] = 177777
2865  010266  010602                    MOV     SP,R2           ;R2 GETS DEST ADDR
2866  010270  005726                    TST     (SP)+           ;RESET SP
2867  010272  000257                    CCC                     ;SCOPE SYNC
2868
2869  010274  112346              2$:   MOVB    (R3)+,-(SP)     ;TEST THE MOVB
2870
2871  010276  022703  064555            CMP     #DBTA+1,R3      ;DID MOVB INCREMENT SRC REG?
2872  010302  001402                    BEQ     4$              ;BR IF YES
2873
2874  010304  000000              3$:   HALT                    ;MOVB FAILED TO UPDATE SRC REG
2875  010306  000762                    BR      1$              ;LOCK ON HARD ERROR
2876
2877  010310  020412              4$:   CMP     R4,(R2)         ;RESULT OK?
2878  010312  001402                    BEQ     6$              ;BR IF YES
2879
2880  010314  000000                    HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
2881  010316  000756                    BR      1$              ;LOCK ON HARD ERROR
2882
2883  010320  020206              6$:   CMP     R2,SP           ;DID SP GET PUSHED BY 2 ?
2884  010322  001402                    BEQ     8$              ;BR IF YES
2885
2886  010324  000000                    HALT                    ;MOVB FAILED TO PUSH SP PROPERLY
2887  010326  000752                    BR      1$              ;LOCK ON HARD ERROR
2888
2889  010330  010506              8$:   MOV     R5,SP           ;RESET SP IN CASE OF ERROR
2890
2891                              ;;*****************************************************************
2892                              ;*TEST 115      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR ODD
2893                              ;;*****************************************************************
2894  010332                      TST115:
2895  010332  012700  000115            MOV     #115,R0         ;;LOAD R0 WITH TEST NUMBER
2896  010336  010605                    MOV     SP,R5           ;SAVE SP
2897  010340  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
2898  010344  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
2899  010346  012703  063761            MOV     #DWTB+3,R3      ;SRC ADDR = DWTB+3
```

```
2900  010352  012746  177777          MOV     #-1,-(SP)       ;[DEST] = 177777
2901  010356  010602                  MOV     SP,R2           ;R2 GETS DEST ADDR
2902  010360  005726                  TST     (SP)+           ;RESET SP
2903  010362  000257                  CCC                     ;SCOPE SYNC
2904
2905  010364  112346          2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
2906
2907  010366  022703  063762          CMP     #DWTB+4,R3      ;DID MOVB INCREMENT SRC REG?
2908  010372  001402                  BEQ     4$              ;BR IF YES
2909
2910  010374  000000          3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
2911  010376  000762                  BR      1$              ;LOCK ON HARD ERROR
2912
2913  010400  020412          4$:     CMP     R4,(R2)         ;RESULT OK?
2914  010402  001402                  BEQ     6$              ;BR IF YES
2915
2916  010404  000000                  HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
2917  010406  000756                  BR      1$              ;LOCK ON HARD ERROR
2918
2919  010410  020206          6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2
2920  010412  001402                  BEQ     8$              ;BR IF YES
2921
2922  010414  000000                  HALT                    ;MOVB FAILED TO PUSH SP
2923  010416  000752                  BR      1$              ;LOCK ON HARD ERROR
2924
2925  010420  010506          8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
2926
2927                          ;;**********************************************************************
2928                          ;*TEST 116      BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST EVEN
2929                          ;;**********************************************************************
2930  010422                  TST116:
2931  010422  012700  000116          MOV     #116,R0         ;;LOAD R0 WITH TEST NUMBER
2932  010426  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2933  010432  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
2934  010436  012705  063756          MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
2935  010442  005012          1$:     CLR     (R2)            ;[DEST] = 000000
2936  010444  000257                  CCC                     ;SCOPE SYNC
2937
2938  010446  116537  000006  063236  2$:     MOVB    6(R5),@#MBUF0   ;TEST THE MOVB
2939
2940  010454  020412                  CMP     R4,(R2)         ;RESULT OK?
2941  010456  001402                  BEQ     TST117          ;;BR IF YES
2942
2943  010460  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2944  010462  000767                  BR      1$              ;LOCK ON HARD ERRROR
2945                          ;;**********************************************************************
2946                          ;*TEST 117      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST EVEN
2947                          ;;**********************************************************************
2948  010464                  TST117:
2949  010464  012700  000117          MOV     #117,R0         ;;LOAD R0 WITH TEST NUMBER
2950  010470  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2951  010474  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
2952  010500  012705  064560          MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
2953  010504  005012          1$:     CLR     (R2)            ;[DEST] = 000000
2954  010506  000257                  CCC                     ;SCOPE SYNC
2955
```

```
2956  010510  116537  000001  063236  2$:     MOVB    1(R5),@#MBUF0   ;TEST THE MOVB
2957
2958  010516  020412                          CMP     R4,(R2)         ;RESULT OK?
2959  010520  001402                          BEQ     TST120          ;;BR IF YES
2960
2961  010522  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2962  010524  000767                          BR      1$              ;LOCK ON HARD ERRROR
2963                                  ;;**********************************************************************
2964                                  ;*TEST 120      BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST ODD
2965                                  ;;**********************************************************************
2966  010526                          TST120:
2967  010526  012700  000120                  MOV     #120,R0         ;;LOAD R0 WITH TEST NUMBER
2968  010532  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2969  010536  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
2970  010542  012705  063756                  MOV     #DWTB,R5        ;BASE SRC ADDR = DWTB
2971  010546  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
2972  010550  000257                          CCC                     ;SCOPE SYNC
2973
2974  010552  116537  000006  063237  2$:     MOVB    6(R5),@#MBUF0+1 ;TEST THE MOVB
2975
2976  010560  020412                          CMP     R4,(R2)         ;RESULT OK?
2977  010562  001402                          BEQ     TST121          ;;BR IF YES
2978
2979  010564  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
2980  010566  000767                          BR      1$              ;LOCK ON HARD ERRROR
2981                                  ;;**********************************************************************
2982                                  ;*TEST 121      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST ODD
2983                                  ;;**********************************************************************
2984  010570                          TST121:
2985  010570  012700  000121                  MOV     #121,R0         ;;LOAD R0 WITH TEST NUMBER
2986                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT2
2987  010574  032737  000004  063160          BIT     #BIT2,@#BPTLOC  ;BREAKPOINT HALT SET ??
2988  010602  001401                          BEQ     .+4             ;BR IF NOT
2989  010604  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2990  010606  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2991  010612  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
2992  010616  012705  064560                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
2993  010622  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
2994  010624  000257                          CCC                     ;SCOPE SYNC
2995
2996  010626  116537  000001  063237  2$:     MOVB    1(R5),@#MBUF0+1 ;TEST THE MOVB
2997
2998  010634  020412                          CMP     R4,(R2)         ;RESULT OK?
2999  010636  001402                          BEQ     TST122          ;;BR IF YES
3000
3001  010640  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3002  010642  000767                          BR      1$              ;LOCK ON HARD ERRROR
3003
3004                                  ;;**********************************************************************
3005                                  ;*TEST 122      BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG=0
3006                                  ;;**********************************************************************
3007  010644                          TST122:
3008  010644  012700  000122                  MOV     #122,R0         ;;LOAD R0 WITH TEST NUMBER
3009  010650  000257                  1$:     CCC                     ;CLEAR ALL FLAGS
3010
3011  010652  001404                  2$:     BEQ     3$              ;NO BR SHOULD OCCUR-FLAG=0
```

```
3012  010654  100403                  BMI     3$              ;NO BR SHOULD OCCUR-FLAG=0
3013  010656  102402                  BVS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3014  010660  103401                  BCS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3015  010662  000402                  BR      TST123          ;;GO TO NEXT TEST
3016
3017  010664  000000          3$:     HALT                    ;ONE OF ABOVE BR'S FAILED
3018  010666  000770                  BR      1$              ;ERROR LOOP RETURN
3019
3020                          ;;**********************************************************************
3021                          ;*TEST 123       BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLAG=1
3022                          ;;**********************************************************************
3023  010670                  TST123:
3024  010670  012700  000123                 MOV     #123,R0         ;;LOAD R0 WITH TEST NUMBER
3025  010674  000277          1$:     SCC                     ;MAKE N:C = 1111
3026
3027  010676  001402          21$:    BEQ     22$             ;TEST THE BEQ-IT SHOULD BR
3028
3029  010700  000000          3$:     HALT                    ;BEQ FAILED
3030  010702  000774                  BR      1$              ;ERROR LOOP RETURN
3031
3032  010704  100402          22$:    BMI     23$             ;TEST THE BMI-IT SHOULD BR
3033
3034  010706  000000          5$:     HALT                    ;BMI FAILED
3035  010710  000771                  BR      1$              ;ERROR LOOP RETURN
3036
3037  010712  102402          23$:    BVS     24$             ;TEST THE BVS-IT SHOULD BR
3038
3039  010714  000000          7$:     HALT                    ;BVS FAILED
3040  010716  000766                  BR      1$              ;ERROR LOOP RETURN
3041
3042  010720                  24$:
3043  010720  103402                  BCS     TST124          ;;TEST THE BCS-IT SHOULD BR
3044
3045  010722  000000          9$:     HALT                    ;BCS FAILED
3046  010724  000763                  BR      1$              ;ERROR LOOP RETURN
3047
3048                          ;;**********************************************************************
3049                          ;*TEST 124       BASIC BVC TEST WITH V=1
3050                          ;;**********************************************************************
3051  010726                  TST124:
3052  010726  012700  000124                 MOV     #124,R0         ;;LOAD R0 WITH TEST NUMBER
3053
3054  010732  000262          1$:     SEV                     ;MAKE V=1
3055
3056  010734  102001          2$:     BVC     3$              ;TEST THE BVC-IT SHOULDN'T BR
3057  010736  000402                  BR      TST125          ;;GO TO NEXT TEST
3058
3059  010740  000000          3$:     HALT                    ;BVC FAILED
3060  010742  000773                  BR      1$              ;ERROR LOOP RETURN
3061
3062                          ;;**********************************************************************
3063                          ;*TEST 125       BASIC BVC TEST WITH V=0
3064                          ;;**********************************************************************
3065  010744                  TST125:
3066  010744  012700  000125                 MOV     #125,R0         ;;LOAD R0 WITH TEST NUMBER
3067
```

```
3068  010750  000242                  1$:     CLV                     ;MAKE V=0
3069
3070  010752                          2$:
3071  010752  102002                          BVC     TST126          ;;TEST THE BVC-IT SHOULD BR
3072
3073  010754  000000                  3$:     HALT                    ;BVC FAILED
3074  010756  000774                          BR      1$              ;ERROR LOOP RETURN
3075
3076                                  ;;**********************************************************************
3077                                  ;*TEST 126      BASIC BGE TEST WITH N,V = 00
3078                                  ;;**********************************************************************
3079  010760                          TST126:
3080  010760  012700  000126                 MOV     #126,R0         ;;LOAD R0 WITH TEST NUMBER
3081
3082  010764  000257                  1$:     CCC                     ;MAKE N:C = 0000
3083
3084  010766                          2$:
3085  010766  002002                          BGE     TST127          ;;TEST THE BGE-IT SHOULD BR
3086
3087  010770  000000                  3$:     HALT                    ;BGE FAILED
3088  010772  000774                          BR      1$              ;ERROR LOOP RETURN
3089
3090                                  ;;**********************************************************************
3091                                  ;*TEST 127      BASIC BGE TEST WITH N,V = 01
3092                                  ;;**********************************************************************
3093  010774                          TST127:
3094  010774  012700  000127                 MOV     #127,R0         ;;LOAD R0 WITH TEST NUMBER
3095
3096  011000  000257                  1$:     CCC                     ;CLEAR FLAGS
3097  011002  000262                          SEV                     ;MAKE N,V = 01
3098
3099  011004  002001                  2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3100  011006  000402                          BR      TST130          ;;GO TO NEXT TEST
3101
3102  011010  000000                  3$:     HALT                    ;BGE FAILED
3103  011012  000772                          BR      1$              ;ERROR LOOP RETURN
3104
3105                                  ;;**********************************************************************
3106                                  ;*TEST 130      BASIC BGE TEST WITH N,V = 10
3107                                  ;;**********************************************************************
3108  011014                          TST130:
3109  011014  012700  000130                 MOV     #130,R0         ;;LOAD R0 WITH TEST NUMBER
3110
3111  011020  000257                  1$:     CCC                     ;CLEAR FLAGS
3112  011022  000270                          SEN                     ;MAKE N,V = 10
3113
3114  011024  002001                  2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3115  011026  000402                          BR      TST131          ;;GO TO NEXT TEST
3116
3117  011030  000000                  3$:     HALT                    ;BGE FAILED
3118  011032  000772                          BR      1$              ;ERROR LOOP RETURN
3119
3120                                  ;;**********************************************************************
3121                                  ;*TEST 131      BASIC BGE TEST WITH N,V = 11
3122                                  ;;**********************************************************************
3123  011034                          TST131:
```

```
3124  011034  012700  000131                MOV     #131,R0         ;;LOAD R0 WITH TEST NUMBER
3125
3126  011040  000257                1$:     CCC                     ;CLEAR FLAGS
3127  011042  000272                        272                     ;MAKE N,V = 11
3128
3129  011044                        2$:
3130  011044  002002                        BGE     TST132          ;;TEST THE BGE-IT SHOULD BR
3131
3132  011046  000000                3$:     HALT                    ;BGE FAILED
3133  011050  000773                        BR      1$              ;ERROR LOOP RETURN
3134
3135                                ;;**********************************************************************
3136                                ;*TEST 132      BASIC BLT TEST WITH N,V = 00
3137                                ;;**********************************************************************
3138  011052                        TST132:
3139  011052  012700  000132                MOV     #132,R0         ;;LOAD R0 WITH TEST NUMBER
3140
3141  011056  000257                1$:     CCC                     ;CLEAR FLAGS
3142
3143  011060  002401                2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3144  011062  000402                        BR      TST133          ;;GO TO NEXT TEST
3145
3146  011064  000000                3$:     HALT                    ;BLT FAILED
3147  011066  000773                        BR      1$              ;ERROR LOOP RETURN
3148
3149                                ;;**********************************************************************
3150                                ;*TEST 133      BASIC BLT TEST WITH N,V = 01
3151                                ;;**********************************************************************
3152  011070                        TST133:
3153  011070  012700  000133                MOV     #133,R0         ;;LOAD R0 WITH TEST NUMBER
3154
3155  011074  000257                1$:     CCC                     ;CLEAR FLAGS
3156  011076  000262                        SEV                     ;MAKE N,V = 01
3157
3158  011100                        2$:
3159  011100  002402                        BLT     TST134          ;;TEST THE BLT-IT SHOULD BR
3160
3161  011102  000000                3$:     HALT                    ;BLT FAILED
3162  011104  000773                        BR      1$              ;ERROR LOOP RETURN
3163
3164                                ;;**********************************************************************
3165                                ;*TEST 134      BASIC BLT TEST WITH N,V = 10
3166                                ;;**********************************************************************
3167  011106                        TST134:
3168  011106  012700  000134                MOV     #134,R0         ;;LOAD R0 WITH TEST NUMBER
3169
3170  011112  000257                1$:     CCC                     ;CLEAR FLAGS
3171  011114  000270                        SEN                     ;SET N - N,V = 10
3172
3173  011116                        2$:
3174  011116  002402                        BLT     TST135          ;;TEST THE BLT-IT SHOULD BR
3175
3176  011120  000000                3$:     HALT                    ;BLT FAILED
3177  011122  000773                        BR      1$              ;ERROR LOOP RETURN
3178
3179                                ;;**********************************************************************
```

```
3180                                    ;*TEST 135      BASIC BLT TEST WITH N,V = 11
3181                                    ;;*****************************************************************
3182  011124                            TST135:
3183  011124  012700  000135                   MOV     #135,R0         ;;LOAD R0 WITH TEST NUMBER
3184
3185  011130  000257                    1$:     CCC                     ;CLEAR FLAGS
3186  011132  000272                            272                     ;MAKE N,V = 11
3187
3188  011134  002401                    2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3189  011136  000402                            BR      TST136          ;;GO TO NEXT TEST
3190
3191  011140  000000                    3$:     HALT                    ;BLT FAILED
3192  011142  000772                            BR      1$              ;ERROR LOOP RETURN
3193
3194                                    ;;*****************************************************************
3195                                    ;*TEST 136      BASIC BGT TEST WITH Z = 1 AND N,V = 01
3196                                    ;;*****************************************************************
3197  011144                            TST136:
3198  011144  012700  000136                   MOV     #136,R0         ;;LOAD R0 WITH TEST NUMBER
3199
3200  011150  000257                    1$:     CCC                     ;CLEAR FLAGS
3201  011152  000266                            266                     ;SET Z AND V
3202
3203  011154  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3204  011156  000402                            BR      TST137          ;;GO TO NEXT TEST
3205
3206  011160  000000                    3$:     HALT                    ;BGT FAILED
3207  011162  000772                            BR      1$              ;ERROR LOOP RETURN
3208
3209                                    ;;*****************************************************************
3210                                    ;*TEST 137      BASIC BGT TEST WITH Z = 0 AND N,V = 01
3211                                    ;;*****************************************************************
3212  011164                            TST137:
3213  011164  012700  000137                   MOV     #137,R0         ;;LOAD R0 WITH TEST NUMBER
3214
3215  011170  000257                    1$:     CCC                     ;CLEAR FLAGS
3216  011172  000262                            SEV                     ;SET V
3217
3218  011174  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3219  011176  000402                            BR      TST140          ;;GO TO SCOPE LOOP EXIT
3220
3221  011200  000000                    3$:     HALT                    ;BGT FAILED
3222  011202  000772                            BR      1$              ;ERROR LOOP RETURN
3223
3224                                    ;;*****************************************************************
3225                                    ;*TEST 140      BASIC BGT TEST WITH Z = 1 AND N,V = 00
3226                                    ;;*****************************************************************
3227  011204                            TST140:
3228  011204  012700  000140                   MOV     #140,R0         ;;LOAD R0 WITH TEST NUMBER
3229
3230  011210  000257                    1$:     CCC                     ;CLEAR FLAGS
3231  011212  000264                            SEZ                     ;SET Z
3232
3233  011214  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3234  011216  000402                            BR      TST141          ;;GO TO SCOPE LOOP EXIT
3235
```

```
3236  011220  000000                3$:     HALT                    ;BGT FAILED
3237  011222  000772                        BR      1$              ;ERROR LOOP RETURN
3238
3239                                ;;**********************************************************************
3240                                ;*TEST 141       BASIC BGT TEST WITH Z = 0 AND N,V = 00
3241                                ;;**********************************************************************
3242  011224                        TST141:
3243  011224  012700  000141               MOV     #141,R0         ;;LOAD R0 WITH TEST NUMBER
3244
3245  011230  000257                1$:     CCC                     ;CLEAR FLAGS
3246
3247  011232                        2$:
3248  011232  003002                        BGT     TST142          ;;TEST THE BGT - IT SHOULD BR
3249
3250  011234  000000                3$:     HALT                    ;BGT FAILED
3251  011236  000774                        BR      1$              ;ERROR LOOP RETURN
3252
3253                                ;;**********************************************************************
3254                                ;*TEST 142       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3255                                ;;**********************************************************************
3256  011240                        TST142:
3257  011240  012700  000142               MOV     #142,R0         ;:LOAD R0 WITH TEST NUMBER
3258
3259  011244  000257                1$:     CCC                     ;CLEAR FLAGS
3260  011246  000266                        266                     ;MAKE N,V = 01 AND Z = 1
3261
3262  011250  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3263  011252  000402                        BR      TST143          ;;GO TO NEXT TEST
3264
3265  011254  000000                3$:     HALT                    ;BGT FAILED
3266  011256  000772                        BR      1$              ;ERROR LOOP RETURN
3267
3268                                ;;**********************************************************************
3269                                ;*TEST 143       BASIC BGT TEST WITH Z = 1 AND N,V = 10
3270                                ;;**********************************************************************
3271  011260                        TST143:
3272  011260  012700  000143               MOV     #143,R0         ;;LOAD R0 WITH TEST NUMBER
3273
3274  011264  000257                1$:     CCC                     ;CLEAR FLAGS
3275  011266  000274                        274                     ;MAKE Z = 1 AND N,V = 10
3276
3277  011270  003001                2$:     BGT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3278  011272  000402                        BR      TST144          ;;GO TO NEXT TEST
3279
3280  011274  000000                3$:     HALT                    ;BLT FAILED
3281  011276  000772                        BR      1$              ;ERROR LOOP RETURN
3282
3283                                ;;**********************************************************************
3284                                ;*TEST 144       BASIC BGT TEST WITH Z = 1 AND N,V = 11
3285                                ;;**********************************************************************
3286  011300                        TST144:
3287  011300  012700  000144               MOV     #144,R0         ;;LOAD R0 WITH TEST NUMBER
3288
3289  011304  000257                1$:     CCC                     ;CLEAR FLAGS
3290  011306  000276                        276                     ;MAKE Z = 1 AND N,V = 11
3291
```

```
3292  011310  003001                  2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3293  011312  000402                          BR      TST145          ;;GO TO NEXT TEST
3294
3295  011314  000000                  3$:     HALT                    ;BLT FAILED
3296  011316  000772                          BR      1$              ;ERROR LOOP RETURN
3297
3298                                  ;;**********************************************************************
3299                                  ;*TEST 145       BASIC BGT TEST WITH Z=0 AND N,V=11
3300                                  ;;**********************************************************************
3301  011320                          TST145:
3302  011320  012700  000145                MOV     #145,R0         ;;LOAD R0 WITH TEST NUMBER
3303
3304  011324  000257                  1$:     CCC                     ;CLEAR FLAGS
3305  011326  000272                          272                     ;MAKE N:C=1010
3306
3307  011330                          2$:
3308  011330  003002                        BGT     TST146          ;;TEST THE BGT - IT SHOULD BR
3309
3310  011332  000000                  3$:     HALT                    ;BGT FAILED
3311  011334  000773                          BR      1$              ;ERROR LOOP RETURN
3312
3313                                  ;;**********************************************************************
3314                                  ;*TEST 146       BASIC BHI TEST WITH Z,C = 00
3315                                  ;;**********************************************************************
3316  011336                          TST146:
3317  011336  012700  000146                MOV     #146,R0         ;;LOAD R0 WITH TEST NUMBER
3318
3319  011342  000257                  1$:     CCC                     ;MAKE Z,C = 00
3320
3321  011344                          2$:
3322  011344  101002                        BHI     TST147          ;;TEST THE BHI-IT SHOULD BR
3323
3324  011346  000000                  3$:     HALT                    ;BHI FAILED
3325  011350  000774                          BR      1$              ;ERROR LOOP RETURN
3326
3327                                  ;;**********************************************************************
3328                                  ;*TEST 147       BASIC BHI TEST WITH Z,C = 01
3329                                  ;;**********************************************************************
3330  011352                          TST147:
3331  011352  012700  000147                MOV     #147,R0         ;;LOAD R0 WITH TEST NUMBER
3332
3333  011356  000257                  1$:     CCC                     ;CLEAR FLAGS
3334  011360  000261                          SEC                     ;MAKE Z,C = 01
3335
3336  011362  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
3337  011364  000402                          BR      TST150          ;;GO TO NEXT TEST
3338
3339  011366  000000                  3$:     HALT                    ;BHI FAILED
3340  011370  000772                          BR      1$              ;ERROR LOOP RETURN
3341
3342                                  ;;**********************************************************************
3343                                  ;*TEST 150       BASIC BHI TEST WITH Z,C = 10
3344                                  ;;**********************************************************************
3345  011372                          TST150:
3346  011372  012700  000150                MOV     #150,R0         ;;LOAD R0 WITH TEST NUMBER
3347
```

```
3348  011376  000257                  1$:     CCC                     ;CLEAR FLAGS
3349  011400  000264                          SEZ                     ;MAKE Z,C = 10
3350
3351  011402  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
3352  011404  000402                          BR      TST151          ;;GO TO NEXT TEST
3353
3354  011406  000000                  3$:     HALT                    ;BHI FAILED
3355  011410  000772                          BR      1$              ;ERROR LOOP RETURN
3356
3357                                  ;;**************************************************************
3358                                  ;*TEST 151      BASIC BHI TEST WITH Z,C = 11
3359                                  ;;**************************************************************
3360  011412                          TST151:
3361  011412  012700  000151                 MOV     #151,R0         ;;LOAD R0 WITH TEST NUMBER
3362
3363  011416  000257                  1$:     CCC                     ;CLEAR FLAGS
3364  011420  000265                          265                     ;MAKE Z,C = 11
3365
3366  011422  101001                  2$:     BHI     3$              ;TEST THE BHI-IT SHOULDN'T BR
3367  011424  000402                          BR      TST152          ;;GO TO NEXT TEST
3368
3369  011426  000000                  3$:     HALT                    ;BHI FAILED
3370  011430  000772                          BR      1$              ;ERROR LOOP RETURN
3371
3372                                  ;;**************************************************************
3373                                  ;*TEST 152      BASIC NEG MODE 0 TEST : [DEST] GT 0
3374                                  ;;**************************************************************
3375  011432                          TST152:
3376  011432  012700  000152                 MOV     #152,R0         ;;LOAD R0 WITH TEST NUMBER
3377  011436  012704  177776                 MOV     #-2,R4          ;RESULT S / B = 177776
3378  011442  012703  000002          1$:    MOV     #2,R3           ;INITIAL [DEST] = 2
3379  011446  000257                          CCC                     ;CLEAR FLAGS
3380  011450  000266                          266                     ;MAKE N:C = 0110
3381
3382  011452  005403                  2$:     NEG     R3              ;TEST THE NEG
3383
3384  011454  100003                          BPL     3$
3385  011456  001402                          BEQ     3$              ;DID N:C = 1001?
3386  011460  102401                          BVS     3$
3387  011462  103402                          BCS     4$
3388
3389  011464  000000                  3$:     HALT                    ;NEGATE FAILED TO ALTER CODES PROPERLY
3390  011466  000765                          BR      1$              ;ERROR LOOP RETURN
3391
3392  011470  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
3393  011472  001402                          BEQ     TST153          ;;BR IF YES
3394
3395  011474  000000                  5$:     HALT                    ;NEG DELIVERED WRONG RESULT
3396  011476  000761                          BR      1$              ;ERROR LOOP RETURN
3397
3398                                  ;;**************************************************************
3399                                  ;*TEST 153      BASIC "SUB #,@#" TEST
3400                                  ;;**************************************************************
3401  011500                          TST153:
3402  011500  012700  000153                 MOV     #153,R0         ;;LOAD R0 WITH TEST NUMBER
3403  011504  012704  000002                 MOV     #2,R4           ;RESULT S / B = 2
```

```
3404  011510  012702  063236          MOV     #MBUF0,R2       ;R2 POINTS TO DEST
3405  011514  012712  000004    1$:   MOV     #4,(R2)         ;INITIAL [DEST] = 4
3406  011520  000257                  CCC                     ;CLEAR FLAGS
3407
3408  011522  162737  000002  063236 2$:   SUB     #2,@#MBUF0      ;TEST THE SUB
3409
3410  011530  020412                  CMP     R4,(R2)         ;RESULT=2?
3411  011532  001403                  BEQ     TST154          ;;BR IF YES
3412  011534  011203                  MOV     (R2),R3         ;GET WAS DATA
3413  011536  000000            3$:   HALT                    ;SUB DELIVERED WRONG RESULT
3414  011540  000765                  BR      1$              ;ERROR LOOP RETURN
3415
3416                            ;;**********************************************************************
3417                            ;*TEST 154      BASIC "SUB @#,RN" TEST
3418                            ;;**********************************************************************
3419  011542                    TST154:
3420  011542  012700  000154          MOV     #154,R0         ;;LOAD R0 WITH TEST NUMBER
3421  011546  012737  000002  063236  MOV     #2,@#MBUF0      ;SRC = 2
3422  011554  012703  000004    1$:   MOV     #4,R3           ;INITIAL [DEST] = 4
3423  011560  000257                  CCC                     ;SCOPE SYNC
3424
3425  011562  163703  063236    2$:   SUB     @#MBUF0,R3      ;TEST THE SUB
3426
3427  011566  020403                  CMP     R4,R3           ;RESULT=2?
3428  011570  001402                  BEQ     TST155          ;;BR IF YES
3429
3430  011572  000000            3$:   HALT                    ;SUB DELIVERED WRONG RESULT
3431  011574  000767                  BR      1$              ;ERROR LOOP RETURN
3432
3433                            ;;**********************************************************************
3434                            ;*TEST 155      BASIC "RTS RN" TEST - <N:C> = 1111
3435                            ;;**********************************************************************
3436  011576                    TST155:
3437  011576  012700  000155          MOV     #155,R0         ;;LOAD R0 WITH TEST NUMBER
3438  011602  010605                  MOV     SP,R5           ;SAVE SP
3439  011604  010506            1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3440  011606  012703  011626          MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
3441  011612  012746  177777          MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 FROM STACK
3442  011616  000277                  SCC                     ;N:C = 1111
3443
3444  011620  000203            2$:   RTS     R3              ;TEST THE RTS - GO TO 4$
3445
3446  011622  000000            3$:   HALT                    ;RTS FAILED TO LOAD THE PC
3447  011624  000767                  BR      1$              ;LOCK ON ERROR
3448
3449  011626  100003            4$:   BPL     5$              ;N:C = 1111 ?
3450  011630  001002                  BNE     5$
3451  011632  102001                  BVC     5$
3452  011634  103402                  BCS     6$
3453
3454  011636  000000            5$:   HALT                    ;RTS ALTERED CODES - CLEARED ONE
3455  011640  000761                  BR      1$              ;LOCK ON ERROR
3456
3457  011642  020327  177777    6$:   CMP     R3,#-1          ;DID R3 GET LOADED FROM STACK ?
3458  011646  001402                  BEQ     8$              ;BR IF YES
3459
```

```
3460  011650  000000                  7$:     HALT                    ;RTS FAILED TO LOAD REG
3461  011652  000754                          BR      1$              ;LOCK ON ERROR
3462
3463  011654  020506                  8$:     CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
3464  011656  001402                          BEQ     TST156          ;;BR IF YES
3465
3466  011660  000000                  9$:     HALT                    ;RTS FAILED TO POP SP
3467  011662  000750                          BR      1$              ;LOCK ON ERROR
3468
3469                                  ;;*****************************************************************
3470                                  ;*TEST 156      BASIC "RTS  PC" TEST
3471                                  ;;*****************************************************************
3472  011664                          TST156:
3473  011664  012700  000156                 MOV     #156,R0         ;;LOAD R0 WITH TEST NUMBER
3474  011670  010605                          MOV     SP,R5           ;SAVE THE ORIGINAL SP
3475  011672  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3476  011674  012746  011710                 MOV     #4$,-(SP)       ;PUSH NEW PC ON STACK
3477  011700  000257                          CCC                     ;SCOPE SYNC
3478
3479  011702  000207                  2$:     RTS     PC              ;TEST THE RTS - GO TO 4$
3480
3481  011704  000000                  3$:     HALT                    ;RTS FAILED TO LOAD PC
3482  011706  000771                          BR      1$              ;LOCK ON HARD ERROR
3483
3484  011710  020605                  4$:     CMP     SP,R5           ;DID SP GET POPPED ?
3485  011712  001402                          BEQ     TST157          ;;BR IF YES
3486
3487  011714  000000                  5$:     HALT                    ;RTS FAILED TO UPDATE SP
3488  011716  000765                          BR      1$              ;LOCK ON HARD ERROR
3489
3490                                  ;;*****************************************************************
3491                                  ;*TEST 157      BASIC "JSR  PC,@#A" TEST
3492                                  ;;*****************************************************************
3493  011720                          TST157:
3494  011720  012700  000157                 MOV     #157,R0         ;;LOAD R0 WITH TEST NUMBER
3495                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT3
3496  011724  032737  000010  063160         BIT     #BIT3,@#BPTLOC  ;BREAKPOINT HALT SET ??
3497  011732  001401                          BEQ     .+4             ;BR IF NOT
3498  011734  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
3499  011736  010605                          MOV     SP,R5           ;SAVE ORIGINAL SP
3500  011740  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3501  011742  000257                          CCC                     ;SCOPE SYNC
3502
3503  011744  004737  011754          2$:     JSR     PC,@#4$ ;TEST THE JSR - GO TO 4$
3504
3505  011750  000000                  3$:     HALT                    ;JSR FAILED TO LOAD PC
3506  011752  000772                          BR      1$              ;LOCK ON HARD ERROR
3507
3508  011754  022726  011750          4$:     CMP     #3$,(SP)+       ;DID JSR SAVE OLD PC ON STACK ?
3509  011760  001402                          BEQ     TST160          ;;BR IF YES
3510
3511  011762  000000                  5$:     HALT                    ;JSR FAILED TO SAVE OLD PC
3512  011764  000765                          BR      1$              ;LOCK ON HARD ERROR
3513
3514                                  ;;*****************************************************************
3515                                  ;*TEST 160      BASIC "RTI" TEST - N:C=0000
```

```
3516                                    ;;*****************************************************************
3517  011766                            TST160:
3518  011766  012700  000160                   MOV     #160,R0         ;;LOAD R0 WITH TEST NUMBER
3519  011772  010605                           MOV     SP,R5           ;SAVE THE SP
3520  011774  010506                    1$:    MOV     R5,SP           ;RESET THE SP FOR ERROR LOOP
3521  011776  012746  000357                   MOV     #357,-(SP)      ;NEW PSW = 357
3522  012002  012746  012022                   MOV     #4$,-(SP)       ;NEW PC = 4$
3523  012006  005037  177776                   CLR     @#PSW           ;MAKE [PSW] = 000
3524  012012  000257                           CCC                     ;MAKE N:C=0000
3525
3526  012014  000002                    2$:    RTI                     ;TEST THE RTI - GO TO 4$
3527
3528  012016  000000                    3$:    HALT                    ;RTI FAILED TO LOAD PC
3529  012020  000765                           BR      1$              ;LOOP ON HARD ERROR
3530
3531  012022  013702  177776            4$:    MOV     @#PSW,R2        ;SAVE THE [PSW] IN R2
3532  012026  022702  000357                   CMP     #357,R2         ;WAS [PSW] = 357 ?
3533  012032  001404                           BEQ     6$              ;BR IF YES
3534
3535  012034  010237  177776                   MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
3536  012040  000000                    5$:    HALT                    ;RTI FAILED TO LOAD PSW
3537  012042  000754                           BR      1$              ;LOCK ON HARD ERROR
3538
3539  012044  020605                    6$:    CMP     SP,R5           ;DID SP GET UPDATED OK ?
3540  012046  001402                           BEQ     TST161          ;;BR IF YES
3541
3542  012050  000000                    7$:    HALT                    ;RTI FAILED TO UPDATE THE SP
3543  012052  000750                           BR      1$              ;LOCK ON HARD ERROR
3544
3545                                    ;;*****************************************************************
3546                                    ;*TEST 161      BASIC "RTI" TEST WITH N:C=1111
3547                                    ;;*****************************************************************
3548  012054                            TST161:
3549  012054  012700  000161                   MOV     #161,R0         ;;LOAD R0 WITH TEST NUMBER
3550  012060  010605                           MOV     SP,R5           ;SAVE THE SP IN R5
3551  012062  010506                    1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3552  012064  005046                           CLR     -(SP)           ;NEW PSW = 000000
3553  012066  012746  012104                   MOV     #4$,-(SP)       ;NEW PC = 4$
3554  012072  012737  000357  177776           MOV     #357,@#PSW      ;MAKE OLD PSW = 357
3555  012100  000240                           NOP                     ;SCOPE SYNC
3556
3557  012102  000002                    2$:    RTI                     ;TEST THE RTI - GO TO 4$
3558
3559  012104  013702  177776            4$:    MOV     @#PSW,R2        ;GET THE PSW
3560  012110  022702  000000                   CMP     #0,R2           ;WAS [PSW]=000
3561  012114  001404                           BEQ     TST162          ;;BR IF YES
3562
3563  012116  010237  177776                   MOV     R2,@#PSW        ;RESTORE ERROR PSW
3564  012122  000000                    3$:    HALT                    ;RTI FAILED TO CLEAR PSW
3565  012124  000756                           BR      1$              ;LOCK ON HARD ERROR
3566
3567                                    ;;*****************************************************************
3568                                    ;*TEST 162      BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
3569                                    ;;*****************************************************************
3570  012126                            TST162:
3571  012126  012700  000162                   MOV     #162,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
3572  012132  010605                          MOV     SP,R5           ;SAVE THE SP
3573  012134  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3574  012136  012737  012174  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
3575  012144  012737  000357  000022          MOV     #357,@#22
3576  012152  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
3577  012160  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
3578  012164  000257                          CCC                     ;SCOPE SYNC
3579
3580  012166  000004                  2$:     IOT                     ;TEST THE IOT
3581
3582  012170  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
3583  012172  000760                          BR      1$              ;LOCK ON HARD ERROR
3584
3585  012174  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
3586  012200  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
3587  012204  001404                          BEQ     6$              ;BR IF YES
3588
3589  012206  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
3590  012212  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
3591  012214  000747                          BR      1$              ;LOCK ON HARD ERROR
3592
3593  012216  022726  012170          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
3594  012222  001404                          BEQ     8$              ;BR IF YES
3595
3596  012224  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
3597  012230  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
3598  012232  000740                          BR      1$              ;LOCK ON HARD ERROR
3599
3600  012234  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
3601  012236  001404                          BEQ     TST163          ;;BR IF YES
3602
3603  012240  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
3604  012244  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
3605  012246  000732                          BR      1$              ;LOCK ON HARD ERROR
3606
3607                                  ;;**********************************************************************
3608                                  ;*TEST 163      BASIC "IOT" TEST - VERIFY LINKAGE TO SCOPE SERVICE
3609                                  ;;**********************************************************************
3610  012250                          TST163:
3611  012250  012700  000163                  MOV     #163,R0         ;;LOAD R0 WITH TEST NUMBER
3612  012254  010605                          MOV     SP,R5           ;SAVE SP
3613  012256  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3614  012260  005037  063170                  CLR     @#SCOFLG        ;TRAP SERVICE WILL COM "SCOFLG"
3615  012264  012737  061536  000020          MOV     #SCOPEA,@#20    ;SET UP IOT VECTOR
3616  012272  005037  000022                  CLR     @#22
3617  012276  000257                          CCC                     ;SCOPE SYNC
3618
3619  012300  000004                  2$:     SCOPE                   ;TEST THE IOT
3620
3621  012302  005137  063170                  COM     @#SCOFLG        ;SCOFLG SHOULD BECOME 000000
3622  012306  001402                          BEQ     4$              ;BR IF IT DID
3623
3624  012310  000000                  3$:     HALT                    ;IOT FAILED TO LINK TO SCOPE SERVICE
3625  012312  000761                          BR      1$              ;LOCK ON HARD ERROR
3626
3627  012314  010506                  4$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
```

```
                                    ;;*****************************************************************
                                    ;*TEST 164      BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
                                    ;;*****************************************************************
012316                              TST164:
012316  012700  000164                      MOV     #164,R0         ;;LOAD R0 WITH TEST NUMBER
012322  010605                              MOV     SP,R5           ;SAVE THE SP
012324  010506                      1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012326  012737  012364  000020              MOV     #4$,@#20        ;SET UP IOT VECTOR
012334  012737  000357  000022              MOV     #357,@#22
012342  012766  177777  177776              MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
012350  005037  177776                      CLR     @#PSW           ;MAKE [PSW] = 000
012354  000257                              CCC                     ;SCOPE SYNC
012356  000004                      2$:     IOT                     ;TEST THE IOT
012360  000000                      3$:     HALT                    ;IOT FAILED TO LOAD PC
012362  000760                              BR      1$              ;LOCK ON HARD ERROR
012364  013702  177776              4$:     MOV     @#PSW,R2        ;GET THE PSW
012370  022702  000357                      CMP     #357,R2         ;DID IOT LOAD A 357 ?
012374  001404                              BEQ     6$              ;BR IF YES
012376  010237  177776                      MOV     R2,@#PSW        ;RESTORE ERROR PSW
012402  000000                      5$:     HALT                    ;IOT FAILED TO LOAD PSW
012404  000747                              BR      1$              ;LOCK ON HARD ERROR
012406  022726  012360              6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
012412  001404                              BEQ     8$              ;BR IF YES
012414  010237  177776                      MOV     R2,@#PSW        ;RESTORE ERROR PSW
012420  000000                      7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
012422  000740                              BR      1$              ;LOCK ON HARD ERROR
012424  005726                      8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
012426  001404                              BEQ     TST165          ;;BR IF YES
012430  010237  177776                      MOV     R2,@#PSW        ;RESTORE ERROR PSW
012434  000000                      9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
012436  000732                              BR      1$              ;LOCK ON HARD ERROR
                                    ;;*****************************************************************
                                    ;*TEST 165      BASIC IOT TEST - VERIFY LOADING PSW WITH 000
                                    ;;*****************************************************************
012440                              TST165:
012440  012700  000165                      MOV     #165,R0         ;;LOAD R0 WITH TEST NUMBER
012444  010605                              MOV     SP,R5           ;SAVE THE SP
012446  010506                      1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012450  012737  012474  000020              MOV     #4$,@#20        ;SET UP IOT VECTOR
012456  005037  000022                      CLR     @#22            ;
012462  012737  000340  177776              MOV     #340,@#PSW      ;MAKE [PSW] = 340
012470  000277                              SCC                     ;MAKE N:C=1111
012472  000004                      2$:     IOT                     ;TEST THE IOT
012474  013702  177776              4$:     MOV     @#PSW,R2        ;GET THE [PSW]
012500  001404                              BEQ     6$              ;BR IF [PSW] = 000
```

```
3684
3685  012502  010237  177776              MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
3686  012506  000000              3$:     HALT                    ;IOT FAILED TO CLEAR THE PSW
3687  012510  000756                      BR      1$              ;LOCK ON HARD ERROR
3688
3689  012512  010506              6$:     MOV     R5,SP           ;RESET THE SP BEFORE CONTINUING
3690
3691                                ;;*****************************************************************
3692                                ;*TEST 166      BASIC "TRAP" TEST - LINKAGE TO PRINT ROUTINE
3693                                ;;*****************************************************************
3694  012514                        TST166:
3695  012514  012700  000166               MOV     #166,R0         ;;LOAD R0 WITH TEST NUMBER
3696  012520  010605                       MOV     SP,R5           ;SAVE THE SP
3697  012522  010506              1$:      MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3698  012524  012737  062126  000034       MOV     #PRINA,@#34     ;SET UP THE "TRAP" VECTOR
3699  012532  005037  000036              CLR     @#36
3700  012536  005037  063162              CLR     @#PRIFLG        ;INITIALIZE TEST FLAG
3701  012542  000257                       CCC                     ;SCOPE SYNC
3702
3703  012544  104401              2$:     TYPE                    ;TEST THE TRAP
3704
3705  012546  005137  063162              COM     @#PRIFLG        ;SHOULD MAKE [PRIFLG] = 000000
3706  012552  001402                       BEQ     TST167          ;;BR IF IT DID
3707
3708  012554  000000              3$:     HALT                    ;TRAP FAILED TO LINK TO PRINT SERV.
3709  012556  000761                      BR      1$              ;LOCK ON HARD ERROR
3710
3711                                ;;*****************************************************************
3712                                ;*TEST 167      BASIC "EMT" TEST - LINKAGE TO ERROR SERVICE
3713                                ;;*****************************************************************
3714  012560                        TST167:
3715  012560  012700  000167               MOV     #167,R0         ;;LOAD R0 WITH TEST NUMBER
3716  012564  010605                       MOV     SP,R5           ;SAVE THE SP
3717  012566  010506              1$:      MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3718  012570  012737  061764  000030       MOV     #ERRA,@#30      ;SET UP THE EMT VECTOR
3719  012576  005037  000032              CLR     @#32
3720  012602  005037  063164              CLR     @#ERRFLG        ;EMT SERVICE WILL COM [ERRFLG]
3721  012606  000257                       CCC                     ;SCOPE SYNC
3722
3723  012610  104000              2$:     ERROR                   ;TEST THE EMT
3724
3725  012612  005137  063164              COM     @#ERRFLG        ;DID EMT SERV. COM ERRFLG?
3726  012616  001402                       BEQ     TST170          ;;BR IF YES
3727
3728  012620  000000              3$:     HALT                    ;EMT DID NOT LINK PROPERLY
3729  012622  000761                      BR      1$              ;LOCK ON HARD ERROR
3730                                ;;*****************************************************************
3731                                ;*TEST 170      BASIC TEST OF RSVD INSTR. TRAP LINKAGE
3732                                ;;*****************************************************************
3733  012624                        TST170:
3734  012624  012700  000170               MOV     #170,R0         ;;LOAD R0 WITH TEST NUMBER
3735  012630  010605                       MOV     SP,R5           ;SAVE THE SP
3736  012632  012737  061040  000010       MOV     #RSVTST,@#10    ;SET UP RSVD INSTR. TRAP VECTOR
3737  012640  012737  000340  000012       MOV     #340,@#12
3738  012646  010506              1$:      MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3739  012650  005037  063172              CLR     @#RSVFLG        ;INITIALIZE TEST FLAG THAT WILL GET
```

```
3740
3741  012654  000257                          CCC                     ;COMPLEMENTED BY TRAP SERVICE
3742                                                                  ;SCOPE SYNC
3743  012656  000007                  2$:     000007                  ;FORCE RSVD INSTR. TRAP
3744
3745  012660  005137  063172                  COM     @#RSVFLG                ;TEST FLAG SHOULD GO TO 000000
3746  012664  001402                          BEQ     4$              ;BR IF TRAP SPRUNG
3747
3748  012666  000000                  3$:     HALT                    ;RSVD INSTR. TRAP FAILED
3749  012670  000766                          BR      1$              ;LOCK ON HARD ERROR
3750
3751  012672  012737  061046  000010  4$:     MOV     #RSERR,@#10     ;SET UP RSVD INSTR TRAP VECTOR TO POINT
3752  012700  012737  000340  000012          MOV     #340,@#12       ;TO ERROR SERVICE ROUTINE
3753
3754                                  ;;*****************************************************************
3755                                  ;*TEST 171      BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
3756                                  ;;*****************************************************************
3757  012706                          TST171:
3758  012706  012700  000171                  MOV     #171,R0         ;;LOAD R0 WITH TEST NUMBER
3759  012712  010605                          MOV     SP,R5           ;SAVE THE SP
3760  012714  012737  061136  000004          MOV     #BETST,@#4      ;SET UP THE BUS ERROR VECTOR
3761  012722  012737  000340  000006          MOV     #340,@#6
3762  012730  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3763  012732  005037  063174                  CLR     @#BERFLG        ;INITIALIZE TEST FLAG THAT WILL GET
3764                                                                  ;COMPLEMENTED BY TRAP SERVICE
3765  012736  000257                          CCC                     ;SCOPE SYNC
3766
3767  012740  005737  177700          2$:     TST     @#177700        ;FORCE BUS TIMEOUT USING R0 ADDR.
3768
3769  012744  005137  063174                  COM     @#BERFLG        ;TEST FLAG SHOULD GO TO 000000
3770  012750  001402                          BEQ     TST172          ;;BR IF TRAP SPRUNG
3771
3772  012752  000000                  3$:     HALT                    ;BUS ERROR FAILED TO SPRING TRAP
3773  012754  000765                          BR      1$              ;LOCK ON HARD ERROR
3774
3775                                  ;;*****************************************************************
3776                                  ;*TEST 172      BASIC TEST FOR ACCESSING DL11 REGISTERS
3777                                  ;;*****************************************************************
3778  012756                          TST172:
3779  012756  012700  000172                  MOV     #172,R0         ;;LOAD R0 WITH TEST NUMBER
3780  012762  005067  050250                  CLR     MBUF0           ;INIT STALL COUNTER
3781  012766  005367  050244          11$:    DEC     MBUF0           ;COUNT THE TIMER
3782  012772  001375                          BNE     11$             ;BR IF NO TIMEOUT
3783  012774  012737  013034  000004          MOV     #3$,@#4 ;SET UP BUS TIMEOUT VECTOR
3784  013002  012737  000340  000006          MOV     #340,@#6
3785  013010  010605                          MOV     SP,R5           ;SAVE TH SP
3786  013012  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3787  013014  012702  177560                  MOV     #RCSR,R2        ;[R2] = STARTING DL11 ADDR.
3788  013020  000257                          CCC                     ;SCOPE SYNC
3789
3790  013022  005722                  2$:     TST     (R2)+           ;REFERENCE DL11 - RCSR
3791  013024  005722                          TST     (R2)+           ;REFERENCE DL11 - RDBR
3792  013026  005722                          TST     (R2)+           ;REFERENCE DL11 - XCSR
3793  013030  005712                          TST     (R2)            ;REFERENCE DL11 - XDBR
3794
3795  013032  000403                          BR      4$              ;GO TO NEXT TEST
```

```
                                ;;*****************************************************************
                                ;*TEST 173       BASIC TEST OF DL11 - XCSR - READY(1)
                                ;;*****************************************************************
013056                          TST173:
013056  012700  000173                MOV     #173,R0         ;;LOAD R0 WITH TEST NUMBER
013062  012702  177564                MOV     #XCSR,R2        ;DEST ADDR = XCSR
013066  012704  000200                MOV     #200,R4         ;RESULT S / B = 200
013072  005012          1$:   CLR     (R2)            ;CLEAR [DEST]
013074  005001                CLR     R1              ;SET UP TIMEOUT COUNTER
013076  000257                CCC                     ;SCOPE SYNC

013100  020412          2$:   CMP     R4,(R2)         ;TEST READY BIT - IT SHOULD BE SET

013102  001405                BEQ     TST174          ;;BR IF IT WAS
013104  005301                DEC     R1              ;TICK-TOCK GOES THE TIMER
013106  001374                BNE     2$              ;BR IF NOT A TIMEOUT

013110  011203                MOV     (R2),R3         ;GET THE WAS DATA
013112  000000          3$:   HALT                    ;READY BIT IN XCSR FAILED ON A (0)
013114  000766                BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 174       BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
                                ;;*****************************************************************
013116                          TST174:
013116  012700  000174                MOV     #174,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 MAINT. BIT
013134  000257                CCC                     ;SCOPE SYNC

013136  020412          2$:   CMP     R4,(R2)         ;TEST MAINT(0)

013140  001403                BEQ     TST175          ;;BR IF MAINT BIT CLEAR

013142  011203                MOV     (R2),R3         ;GET THE WAS DATA
013144  000000          3$:   HALT                    ;CAN'T CLEAR MAINT BIT
013146  000771                BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 175       BASIC TEST OF DL11 XCSR - MAINT BIT = 1
                                ;;*****************************************************************
013150                          TST175:
013150  012700  000175                MOV     #175,R0         ;;LOAD R0 WITH TEST NUMBER
013154  012702  177564                MOV     #XCSR,R2        ;DEST ADDR = XCSR
013160  012704  000204                MOV     #204,R4         ;RESULT S / B = 204
013164  012712  000004          1$:   MOV     #4,(R2)         ;SET THE MAINT. BIT
013170  000257                CCC                     ;SCOPE SYNC
```

```
3852  013172  020412                  2$:     CMP     R4,(R2)         ;TEST MAINT.(1)
3853
3854  013174  001403                          BEQ     TST176          ;;BR IF IT WAS
3855
3856  013176  011203                          MOV     (R2),R3         ;GET THE WAS DATA
3857  013200  000000                  3$:     HALT                    ;CAN'T SET MAINT BIT IN XCSR
3858  013202  000770                          BR      1$              ;LOCK 0 HARD ERROR
3859
3860                                  ;;**********************************************************************
3861                                  ;*TEST 176      BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
3862                                  ;THIS ROUTINE USES THE MAINTENANCE MODE FEATURE OF THE DL11 TO
3863                                  ;TURN AROUND A STRING OF 8 CHARACTERS TO THE DL11. THIS STRING CONSISTS
3864                                  ;OF ALTERNATING NULL / DELETE CHARS WHICH ARE NON PRINTING. THE 8 CHARS
3865                                  ;ARE OUTPUT THEN READ BACK INTO A CORE BUFFER AND THEN THE INPUT AND
3866                                  ;OUTPUT CORE BUFFERS ARE CHECKED FOR EQUIVALENCE. IF AN ERROR IS DET-
3867                                  ;ECTED DURING THE COMPARISON THE ROUTINE HALTS WITH THE WAS AND S / B
3868                                  ;DATA IN R3 AND R4 RESPECTFULLY. A TIMER IS EMPLOYED TO PREVENT THE
3869                                  ;TEST FROM HANGING IF RECEIVER DONE DOES NOT RESPOND.
3870                                  ;;**********************************************************************
3871  013204                          TST176:
3872  013204  012700  000176                  MOV     #176,R0         ;;LOAD R0 WITH TEST NUMBER
3873  013210  012702  177560          6$:     MOV     #RCSR,R2        ;R2 POINTS TO DL11 - START ADDR
3874  013214  105762  000002                  TSTB    2(R2)           ;REFERENCE DL11 INPUT DATA BUFFER TWICE
3875  013220  105762  000002                  TSTB    2(R2)           ;TO FLUSH RCVR "DONE" BIT
3876  013224  012703  063212                  MOV     #IBUF,R3        ;R3 POINTS TO CORE INPUT BUFFER
3877  013230  012704  063202                  MOV     #OBUF,R4        ;R4 POINTS TO CORE OUTPUT BUFFER
3878  013234  012705  000010                  MOV     #10,R5          ;R5 WILL COUNT 8 CHARS OUTPUT
3879  013240  012762  000004  000004          MOV     #4,4(R2)        ;TURN ON MAINT MODE
3880
3881  013246  005001                  1$:     CLR     R1              ;R1 USED AS TIMEOUT COUNTER
3882  013250  112462  000006                  MOVB    (R4)+,6(R2)     ;LOAD OUTPUT BUFFER IN DL11
3883  013254  105712                  2$:     TSTB    (R2)            ;RECEIVER DONE SET ?
3884  013256  100404                          BMI     3$              ;BR IF YES
3885  013260  005301                          DEC     R1              ;COUNT THE TIMER
3886  013262  001374                          BNE     2$              ;BR IF NO TIMEOUT
3887
3888  013264  000000                          HALT                    ;DL11 FAILED TO RESPOND IN TIME
3889  013266  000750                          BR      6$              ;LOCK ON HARD ERROR
3890
3891  013270  116223  000002          3$:     MOVB    2(R2),(R3)+     ;READ THE DL11 INPUT BUFFER INTO CORE
3892  013274  005305                          DEC     R5              ;COUNT ONE CHAR
3893  013276  001363                          BNE     1$              ;BR IF NOT DONE 8 CHARS
3894
3895  013300  005062  000004                  CLR     4(R2)           ;TURN OFF MAINT. MODE
3896  013304  012705  000010                  MOV     #10,R5          ;RESET CHAR COUNTER
3897  013310  012703  063212                  MOV     #IBUF,R3        ;RESET INBUF POINTER
3898  013314  012704  063202                  MOV     #OBUF,R4        ;RESET OUTBUF POINTER
3899
3900  013320  122324                  4$:     CMPB    (R3)+,(R4)+     ;INPUT = OUTPUT ??
3901  013322  001003                          BNE     5$              ;BR IF NOT
3902  013324  005305                          DEC     R5              ;COUNT ONE CHECKED
3903  013326  001374                          BNE     4$              ;BR UNTIL 8 DONE
3904  013330  000410                          BR      CITST           ;GO TO NEXT TEST
3905
3906  013332  114303                  5$:     MOVB    -(R3),R3        ;WAS DATA IN R3 [BITS 7:0]
3907  013334  114404                          MOVB    -(R4),R4        ;S / B DATA IN R4 [BITS 7:0]
```

```
 3908  013336  042703  177400          BIC     #177400,R3      ;STRIP OFF BITS <15:08>
 3909  013342  042704  177400          BIC     #177400,R4      ;  "     "
 3910  013346  000000                  HALT                    ;RECEIVED DATA NOT EQUAL TO OUTPUT DATA
 3911  013350  000717                  BR      6$              ;LOCK ON HARD ERROR
```

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

013352  012737  061204  000020  CITST:  MOV     #$SCOPE,@#20    ;SET UP IOT VECTOR
013360  005037  000022                  CLR     @#22
013364  012737  061544  000030          MOV     #$ERROR,@#30    ;SET UP EMT VECTOR
013372  012737  000340  000032          MOV     #340,@#32
013400  012737  063112  000034          MOV     #$TRAP,@#34     ;SET UP TRAP VECTOR
013406  012737  000340  000036          MOV     #340,@#36
013414  012737  060610  000024          MOV     #$PWRDN,@#24    ;SET UP POWER FAIL VECTOR
013422  012737  000340  000026          MOV     #340,@#26
013430  105737  001141                  TSTB    @#$ENVM         ;DO NOT SIZE BIT SET?
013434  100003                          BPL     3$              ;BR IF NOT - USE HARDWARE SWITCH REG
013436  012737  001142  001040          MOV     #$SWREG,@#SWR   ;USE APT SWITCH REG.
013444  032777  010000  165366  3$:     BIT     #SW12,@SWR      ;INHIBIT PRINTING INTRO. I.D. MESSAGE?
013452  001007                          BNE     1$              ;BR IF YES
013454  005737  063200                  TST     @#ONCE          ;FIRST TIME INTO "CIT" TESTS ?
013460  001004                          BNE     1$              ;BR IF NOT - PRINT ID ONLY ONCE
013462  005137  063200                  COM     @#ONCE          ;SET FLAG TO INHIBIT PRINTING AGAIN
013466  104401                          TYPE                    ;IDENTIFY THIS PROGRAM
013470  065065                          IDENT1                  ;ADDR OF THE ID MESSAGE
013472  005037  177776          1$:     CLR     @#PSW           ;SET CPU PRIORITY TO LEVEL 000
013476  012737  003272  001006          MOV     #TST0,@#$LPADR  ;INITIALIZE SCOPE LOOP RETURN
013504  012737  000040  001110          MOV     #40,@#$TIMES    ;ITERATE ON BIT SECTION 32 TIMES
013512  010037  001124                  MOV     R0,@#$TESTN     ;PREVENT MISSED TEST ERROR ON
                                                                ;FIRST SCOPE CALL

                                        ;;*****************************************************************
                                        ;*TEST 177      BCC TEST WITH C=1
                                        ;;*****************************************************************
013516                                  TST177:
013516  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013520  012700  000177                          MOV     #177,R0         ;;LOAD R0 WITH TEST NUMBER
013524  013701  013532                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013530  000261                                  SEC                     ;MAKE C=1

013532  103001          2$:     BCC     3$              ;TEST THE BCC, IT SHOULDN'T BR
013534  000401                  BR      TST200          ;;GO TO SCOPE EXIT

013536  104005          3$:     ERROR   5               ;BCC FAILED

                                        ;;*****************************************************************
                                        ;*TEST 200      BCC TEST WITH C=0
                                        ;;*****************************************************************
013540                                  TST200:
013540  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013542  012700  000200                          MOV     #200,R0         ;;LOAD R0 WITH TEST NUMBER
013546  013701  013554                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013552  000241                                  CLC                     ;MAKE C=0

013554                                  2$:
013554  103001                          BCC     TST201          ;;TEST THE BCC-IT SHOULD BR

013556  104005          3$:     ERROR   5               ;BCC FAILED
```

```
                                ;;*****************************************************************
                                ;*TEST 201       VERIFY NO BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013560                          TST201:
013560  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013562  012700  000201                  MOV     #201,R0         ;;LOAD R0 WITH TEST NUMBER
013566  013701  013604                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013572  012704  000017                  MOV     #17,R4          ;S/B PSW
013576  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013602  000277                          SCC                     ;MAKE N:C = 1111

013604  103004                  2$:     BCC     3$              ;TEST THE BCC-IT SHOULDN'T BR

013606  013703  177776                  MOV     @#PSW,R3        ;GET WAS FLAGS
013612  020304                          CMP     R3,R4           ;N:C = 1111?
013614  001401                          BEQ     TST202          ;;BR IF YES

013616  104001                  3$:     ERROR   1               ;NO BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 202       VERIFY BRANCH MICROROUTINE DOES NOT CLR FLAGS
                                ;;*****************************************************************
013620                          TST202:
013620  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013622  012700  000202                  MOV     #202,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  000401                  2$:     BR      4$              ;TEST THE BR

013646  104005                  3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK

013650  013703  177776          4$:     MOV     @#PSW,R3        ;GET THE FLAGS
013654  020304                          CMP     R3,R4           ;N:C = 1111?
013656  001401                          BEQ     TST203          ;;BR IF YES

013660  104001                  5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES

                                ;;*****************************************************************
                                ;*TEST 203       VERIFY NO BRANCH MICROROUTINE DOES NOT SET FLAGS
                                ;;*****************************************************************
013662                          TST203:
013662  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
013664  012700  000203                  MOV     #203,R0         ;;LOAD R0 WITH TEST NUMBER
013670  013701  013704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
013674  005004                          CLR     R4              ;PSW S/B = 0
013676  012702  177776                  MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL

013702  000257                          CCC                     ;MAKE N:C = 0000

013704  103404                  2$:     BCS     3$              ;TEST THE BCS-IT SHOULDN'T BR
```

```
4024  013706  013703  177776              MOV     @#PSW,R3        ;GET FLAGS
4025  013712  005703                      TST     R3              ;N:C = 0000
4026  013714  001401                      BEQ     TST204          ;;BR IF YES
4027
4028  013716  104001              3$:     ERROR   1               ;NO BRANCH MICROROUTINE-ALTERED CODES
4029
4030                              ;;**********************************************************************
4031                              ;*TEST 204      VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
4032                              ;;**********************************************************************
4033  013720                      TST204:
4034  013720  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4035  013722  012700  000204              MOV     #204,R0         ;;LOAD R0 WITH TEST NUMBER
4036  013726  013701  013742              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4037  013732  005004                      CLR     R4              ;PSW S/B = 0
4038  013734  012702  177776              MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL
4039
4040  013740  000257                      CCC                     ;MAKE N:C = 0000
4041
4042  013742  000401              2$:     BR      4$              ;TEST THE BR
4043
4044  013744  104005              3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK
4045
4046  013746  013703  177776      4$:     MOV     @#PSW,R3        ;GET FLAGS
4047  013752  005703                      TST     R3              ;N:C = 0000
4048  013754  001401                      BEQ     TST205          ;;BR IF YES
4049
4050  013756  104001              5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES.
4051
4052                              ;;**********************************************************************
4053                              ;*TEST 205      BLE TEST WITH Z = 0, AND N,V = 00
4054                              ;;**********************************************************************
4055  013760                      TST205:
4056  013760  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4057  013762  012700  000205              MOV     #205,R0         ;;LOAD R0 WITH TEST NUMBER
4058  013766  013701  013774              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4059  013772  000257                      CCC                     ;CLEAR FLAGS
4060
4061  013774  003401              2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
4062  013776  000401                      BR      TST206          ;;GO TO SCOPE EXIT
4063
4064  014000  104005              3$:     ERROR   5               ;BLE FAILED
4065
4066                              ;;**********************************************************************
4067                              ;*TEST 206      BLE TEST WITH Z = 1 AND N,V = 00
4068                              ;;**********************************************************************
4069  014002                      TST206:
4070  014002  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4071  014004  012700  000206              MOV     #206,R0         ;;LOAD R0 WITH TEST NUMBER
4072  014010  013701  014020              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4073  014014  000257                      CCC                     ;CLEAR FLAGS
4074  014016  000264                      SEZ                     ;SET Z = 1
4075
4076  014020                      2$:
4077  014020  003401                      BLE     TST207          ;;TEST THE BLE-IT SHOULD BR
4078
4079  014022  104005              3$:     ERROR   5               ;BLE FAILED
```

```
4080
4081                                    ;;*****************************************************************
4082                                    ;*TEST 207      BLE TEST WITH Z = 0 AND N,V = 01
4083                                    ;;*****************************************************************
4084  014024                            TST207:
4085  014024  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4086  014026  012700  000207                  MOV     #207,R0         ;;LOAD R0 WITH TEST NUMBER
4087  014032  013701  014042                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4088  014036  000257                           CCC                     ;CLEAR FLAGS
4089  014040  000262                           SEV                     ;MAKE Z = 0 AND N,V = 01
4090
4091  014042                            2$:
4092  014042  003401                           BLE     TST210          ;;TEST THE BLE-IT SHOULD BR
4093
4094  014044  104005                    3$:    ERROR   5               ;BLE FAILED
4095
4096                                    ;;*****************************************************************
4097                                    ;*TEST 210      BLE TEST WITH Z = 0 AND N,V = 10
4098                                    ;;*****************************************************************
4099  014046                            TST210:
4100  014046  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4101  014050  012700  000210                  MOV     #210,R0         ;;LOAD R0 WITH TEST NUMBER
4102  014054  013701  014064                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4103  014060  000257                           CCC                     ;CLEAR FLAGS
4104  014062  000270                           SEN                     ;MAKE Z = 0 AND N,V = 10
4105
4106  014064                            2$:
4107  014064  003401                           BLE     TST211          ;;TEST THE BLE-IT SHOULD BR
4108
4109  014066  104005                    3$:    ERROR   5               ;BLE FAILED
4110
4111                                    ;;*****************************************************************
4112                                    ;*TEST 211      BLE TEST WITH Z = 0 AND N,V = 11
4113                                    ;;*****************************************************************
4114  014070                            TST211:
4115  014070  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4116  014072  012700  000211                  MOV     #211,R0         ;;LOAD R0 WITH TEST NUMBER
4117  014076  013701  014106                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4118  014102  000257                           CCC                     ;CLEAR FLAGS
4119  014104  000272                           272                     ;MAKE Z = 0 AND N,V = 11
4120
4121  014106  003401                    2$:    BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
4122  014110  000401                           BR      TST212          ;;GO TO SCOPE EXIT
4123
4124  014112  104005                    3$:    ERROR   5               ;BLE FAILED
4125
4126                                    ;;*****************************************************************
4127                                    ;*TEST 212      BLOS TEST WITH Z,C = 00
4128                                    ;;*****************************************************************
4129  014114                            TST212:
4130  014114  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4131  014116  012700  000212                  MOV     #212,R0         ;;LOAD R0 WITH TEST NUMBER
4132  014122  013701  014130                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4133  014126  000257                           CCC                     ;MAKE Z,C = 00
4134
4135  014130  101401                    2$:    BLOS    3$              ;TEST THE BLOS-IT SHOULDN'T BR
```

```
4136  014132  000401                    BR      TST213          ;;GO TO SCOPE EXIT
4137
4138  014134  104005            3$:     ERROR   5               ;BLOS FAILED
4139
4140                            ;;**********************************************************************
4141                            ;*TEST 213      BLOS TEST WITH Z,C = 01
4142                            ;;**********************************************************************
4143  014136                    TST213:
4144  014136  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4145  014140  012700  000213            MOV     #213,R0         ;;LOAD R0 WITH TEST NUMBER
4146  014144  013701  014154            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4147  014150  000257                    CCC                     ;CLEAR FLAGS
4148  014152  000261                    SEC                     ;MAKE Z,C = 01
4149
4150  014154                    2$:
4151  014154  101401                    BLOS    TST214          ;;TEST THE BLOS-IT SHOULD BR
4152
4153  014156  104005            3$:     ERROR   5               ;BLOS FAILED
4154
4155                            ;;**********************************************************************
4156                            ;*TEST 214      BLOS TEST WITH Z,C = 10
4157                            ;;**********************************************************************
4158  014160                    TST214:
4159  014160  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4160  014162  012700  000214            MOV     #214,R0         ;;LOAD R0 WITH TEST NUMBER
4161  014166  013701  014176            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4162  014172  000257                    CCC                     ;CLEAR FLAGS
4163  014174  000264                    SEZ                     ;MAKE Z,C = 10
4164
4165  014176                    2$:
4166  014176  101401                    BLOS    TST215          ;;TEST THE BLOS-IT SHOULD BR
4167
4168  014200  104005            3$:     ERROR   5               ;BLOS FAILED
4169
4170                            ;;**********************************************************************
4171                            ;*TEST 215      BLOS TEST WITH Z,C = 11
4172                            ;;**********************************************************************
4173  014202                    TST215:
4174  014202  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4175  014204  012700  000215            MOV     #215,R0         ;;LOAD R0 WITH TEST NUMBER
4176  014210  013701  014220            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4177  014214  000257                    CCC                     ;CLEAR FLAGS
4178  014216  000265                    265                     ;MAKE Z,C = 11
4179
4180  014220                    2$:
4181  014220  101401                    BLOS    TST216          ;;TEST THE BLOS-IT SHOULD BR
4182
4183  014222  104005            3$:     ERROR   5               ;BLOS FAILED
4184
4185                            ;;**********************************************************************
4186                            ;*TEST 216      SXT MODE 0 TEST WITH N = 0 AND C = 1
4187                            ;;**********************************************************************
4188  014224                    TST216:
4189  014224  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4190  014226  012700  000216            MOV     #216,R0         ;;LOAD R0 WITH TEST NUMBER
4191  014232  013701  014250            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
4192  014236  005004                          CLR     R4              ;RESULT S / B = 0
4193  014240  012703  177777                 MOV     #-1,R3          ;INITAL DEST. OP = 177777
4194  014244  000257                          CCC                     ;CLEAR CODES
4195  014246  000263                          263                     ;N:C = 0011
4196
4197  014250  006703                  2$:     SXT     R3              ;TEST THE SXT
4198
4199  014252  100403                          BMI     3$
4200  014254  001002                          BNE     3$              ;DID SXT MAKE N:C = 0101?
4201  014256  102401                          BVS     3$
4202  014260  103401                          BCS     4$
4203
4204  014262  104002                  3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4205
4206  014264  005703                  4$:     TST     R3              ;DID RESULT = 0?
4207  014266  001401                          BEQ     TST217          ;;BR IF IT DID
4208
4209  014270  104002                  5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT TO R3
4210
4211                                  ;;*********************************************************************
4212                                  ;*TEST 217       SXT MODE 0 TEST WITH N = 0 AND C = 0
4213                                  ;*********************************************************************
4214  014272                          TST217:
4215  014272  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4216  014274  012700  000217                  MOV     #217,R0         ;;LOAD R0 WITH TEST NUMBER
4217  014300  013701  014326                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4218                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT4
4219  014304  032737  000020  063160          BIT     #BIT4,@#BPTLOC  ;BREAKPOINT HALT SET ??
4220  014312  001401                          BEQ     .+4             ;BR IF NOT
4221  014314  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
4222
4223  014316  005004                          CLR     R4              ;RESULT S / B = 0
4224  014320  012703  177777                 MOV     #-1,R3          ;INITIAL DEST OP = 177777
4225  014324  000257                          CCC                     ;CLEAR N:C
4226
4227  014326  006703                  2$:     SXT     R3              ;TEST THE SXT
4228  014330  103001                          BCC     TST220          ;;BR IF "C" STILL CLEAR
4229
4230  014332  104002                  3$:     ERROR   2               ;SXT AFFECTED "C" BIT
4231
4232                                  ;;*********************************************************************
4233                                  ;*TEST 220       SXT MODE 0 TEST WITH N = 1 AND C = 1
4234                                  ;*********************************************************************
4235  014334                          TST220:
4236  014334  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4237  014336  012700  000220                  MOV     #220,R0         ;;LOAD R0 WITH TEST NUMBER
4238  014342  013701  014356                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4239  014346  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
4240  014352  005003                          CLR     R3              ;INITIAL DEST OP = 0
4241  014354  000277                          SCC                     ;MAKE N:C = 1111
4242
4243  014356  006703                  2$:     SXT     R3              ;TEST THE SXT
4244
4245  014360  100003                          BPL     3$
4246  014362  001402                          BEQ     3$              ;N:C = 1001?
4247  014364  102401                          BVS     3$
```

```
4248  014366  103401                        BCS     4$
4249
4250  014370  104002                3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4251
4252  014372  010305                4$:     MOV     R3,R5           ;GET RESULT
4253  014374  005105                        COM     R5              ;COMPLEMENT IT-SHOULD GO TO 0
4254  014376  001401                        BEQ     TST221          ;;BR IF RESULT OF SXT = 1
4255
4256  014400  104002                5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT.
4257
4258                                ;;**************************************************************
4259                                ;*TEST 221      SXT MODE 0 TEST WITH N = 1 AND C = 0
4260                                ;;**************************************************************
4261  014402                        TST221:
4262  014402  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4263  014404  012700  000221                MOV     #221,R0         ;;LOAD R0 WITH TEST NUMBER
4264  014410  013701  014426                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4265  014414  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
4266  014420  005003                        CLR     R3              ;INITIAL DEST OP = 0
4267  014422  000257                        CCC                     ;CLEAR FLAGS
4268  014424  000276                        276                     ;MAKE N:C = 1110
4269
4270  014426  006703                2$:     SXT     R3              ;TEST THE SXT
4271  014430  103001                        BCC     TST222          ;;BR IF "C" UNAFFECTED
4272
4273  014432  104002                3$:     ERROR   2               ;SXT SET "C" BIT
4274
4275                                ;;**************************************************************
4276                                ;*TEST 222      SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
4277                                ;;**************************************************************
4278  014434                        TST222:
4279  014434  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4280  014436  012700  000222                MOV     #222,R0         ;;LOAD R0 WITH TEST NUMBER
4281  014442  013701  014464                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4282  014446  012702  063236                MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4283  014452  005004                        CLR     R4              ;RESULT S / B = 0
4284  014454  012712  177777                MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4285  014460  000257                        CCC                     ;CLEAR CODES
4286  014462  000263                        263                     ;MAKE N:C = 0011
4287
4288  014464  006712                2$:     SXT     (R2)            ;TEST THE SXT - DM1
4289
4290  014466  100403                        BMI     3$
4291  014470  001002                        BNE     3$              ;N:C = 0101
4292  014472  102401                        BVS     3$
4293  014474  103401                        BCS     4$
4294
4295  014476  104001                3$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
4296
4297  014500  005712                4$:     TST     (R2)            ;DID RESULT = 0?
4298  014502  001401                        BEQ     11$             ;BR IF YES
4299
4300  014504  104001                5$:     ERROR   1               ;SXT SHOULD HAVE ZEROED [DEST]
4301
4302  014506  012702  063236        11$:    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4303  014512  013701  014526                MOV     @#12$,R1        ;LOAD R1 WITH TEST INSTR WORD
```

```
4304  014516  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4305  014522  000257                  CCC                     ;CLEAR CODES
4306  014524  000263                  263                     ;MAKE N:C = 0011
4307
4308  014526  006722          12$:    SXT     (R2)+           ;TEST SXT - DM2
4309
4310  014530  100403                  BMI     7$              ;N:C = 0101 ?
4311  014532  001002                  BNE     7$
4312  014534  102401                  BVS     7$
4313  014536  103401                  BCS     6$
4314
4315  014540  104001          7$:     ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
4316
4317  014542  005737  063236  6$:     TST     @#MBUF0         ;DID RESULT GET ZEROED ?
4318  014546  001401                  BEQ     8$              ;BR IF YES
4319
4320  014550  104001          9$:     ERROR   1               ;SXT FAILED TO ZERO [DEST]
4321
4322  014552  020227  063240  8$:     CMP     R2,#MBUF0+2     ;WAS IT REALLY MODE 2 ?
4323  014556  001401                  BEQ     TST223          ;;BR IF YES
4324
4325  014560  104001                  ERROR   1               ;SXT FAILED TO AUTO INCREMENT
4326
4327                          ;;**************************************************************
4328                          ;*TEST 223      SXT MODE 1 TEST WITH N = 0 AND C = 0
4329                          ;;**************************************************************
4330  014562                  TST223:
4331  014562  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4332  014564  012700  000223          MOV     #223,R0         ;;LOAD R0 WITH TEST NUMBER
4333  014570  013701  014610          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4334  014574  005004                  CLR     R4              ;RESULT S / B = 0
4335  014576  012702  063236          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4336  014602  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4337  014606  000257                  CCC                     ;CLEAR "C" BIT
4338
4339  014610  006712          2$:     SXT     (R2)            ;TEST THE SXT
4340  014612  103001                  BCC     TST224          ;;BR IF "C" UNDISTURBED
4341
4342  014614  104001          3$:     ERROR   1               ;SXT SET THE "C" BIT
4343
4344                          ;;**************************************************************
4345                          ;*TEST 224      SXT MODE 1 TEST WITH N = 1 AND C = 1
4346                          ;;**************************************************************
4347  014616                  TST224:
4348  014616  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4349  014620  012700  000224          MOV     #224,R0         ;;LOAD R0 WITH TEST NUMBER
4350  014624  013701  014644          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4351  014630  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
4352  014634  012702  063236          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4353  014640  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
4354  014642  000277                  SCC                     ;MAKE N:C = 1111
4355
4356  014644  006712          2$:     SXT     (R2)            ;TEST THE SXT
4357
4358  014646  100003                  BPL     3$
4359  014650  001402                  BEQ     3$              ;N:C = 1001?
```

```
4360  014652  102401                          BVS     3$
4361  014654  103401                          BCS     4$
4362
4363  014656  104001                3$:       ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
4364
4365  014660  021204                4$:       CMP     (R2),R4         ;RESULT = 177777?
4366  014662  001401                          BEQ     TST225          ;;BR IF YES
4367
4368  014664  104001                5$:       ERROR   1               ;SXT DELIVERED WRONG RESULT
4369
4370                                ;;**************************************************************
4371                                ;*TEST 225      SXT MODE 1 TEST WITH N = 1 AND C = 0
4372                                ;;**************************************************************
4373  014666                        TST225:
4374  014666  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4375  014670  012700  000225                 MOV     #225,R0         ;;LOAD R0 WITH TEST NUMBER
4376  014674  013701  014716                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4377  014700  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
4378  014704  012702  063236                 MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4379  014710  005012                          CLR     (R2)            ;INITIAL [DEST] = 0
4380  014712  000257                          CCC                     ;CLEAR FLAGS
4381  014714  000276                          276                     ;MAKE N:C = 1110
4382
4383  014716  006712                2$:       SXT     (R2)            ;TEST THE SXT
4384  014720  103001                          BCC     TST226          ;;BR IF "C" UNAFFECTED
4385
4386  014722  104001                3$:       ERROR   1               ;SXT SET THE "C" BIT
4387
4388                                ;;**************************************************************
4389                                ;*TEST 226      SWAB MODE 0 TEST WITH POS. RESULT
4390                                ;;**************************************************************
4391  014724                        TST226:
4392  014724  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4393  014726  012700  000226                 MOV     #226,R0         ;;LOAD R0 WITH TEST NUMBER
4394  014732  013701  014752                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4395  014736  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 177400
4396  014742  012703  000377                 MOV     #377,R3         ;INITIAL DEST OP = 377
4397  014746  000257                          CCC                     ;CLEAR FLAGS
4398  014750  000273                          273                     ;MAKE N:C = 1011
4399
4400  014752  000303                2$:       SWAB    R3              ;TEST THE SWAB
4401
4402  014754  100403                          BMI     3$
4403  014756  001002                          BNE     3$              ;N:C = 0100
4404  014760  102401                          BVS     3$
4405  014762  103001                          BCC     4$
4406
4407  014764  104002                3$:       ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
4408
4409  014766  020403                4$:       CMP     R4,R3           ;CORRECT RESULT?
4410  014770  001401                          BEQ     TST227          ;;BR IF YES
4411
4412  014772  104002                5$:       ERROR   2               ;SWAB DELIVERED WRONG RESULT
4413
4414                                ;;**************************************************************
4415                                ;*TEST 227      SWAB MODE 0 TEST WITH NEG. RESULT
```

```
                                        ;;*****************************************************************
014774                                  ;TST227:
014774  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014776  012700  000227                        MOV     #227,R0         ;;LOAD R0 WITH TEST NUMBER
015002  013701  015022                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
015006  012704  000377                        MOV     #377,R4         ;RESULT S  /  B = 377
015012  012703  177400                        MOV     #177400,R3      ;INITIAL DEST OP = 177400
015016  000257                                CCC                     ;CLEAR FLAGS
015020  000267                                267                     ;MAKE N:C = 0111
                                        
015022  000303                        2$:     SWAB    R3              ;TEST THE SWAB
                                        
015024  100003                                BPL     3$
015026  001402                                BEQ     3$              ;DID SWAB MAKE N:C = 1000
015030  102401                                BVS     3$
015032  103001                                BCC     4$
                                        
015034  104002                        3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
                                        
015036  020403                        4$:     CMP     R4,R3           ;DID SWAB DELIVER CORRECT RESULT?
015040  001401                                BEQ     TST230          ;;BR IF OK
                                        
015042  104002                        5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
                                        
                                        ;;*****************************************************************
                                        ;*TEST 230      SWAB MODE 1 AND 2 TEST WITH POS. RESULT
                                        ;;*****************************************************************
015044                                  TST230:
015044  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
015046  012700  000230                        MOV     #230,R0         ;;LOAD R0 WITH TEST NUMBER
015052  013701  015076                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
015056  012704  177400                        MOV     #177400,R4      ;RESULT S  /  B = 177400
015062  012702  063236                        MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
015066  012712  000377                        MOV     #377,(R2)       ;SET UP DEST OP = 377
015072  000257                                CCC                     ;CLEAR FLAGS
015074  000273                                273                     ;MAKE N:C = 1011
                                        
015076  000312                        2$:     SWAB    (R2)            ;TEST THE SWAB - DMI
                                        
015100  100403                                BMI     3$
015102  001002                                BNE     3$              ;N:C = 0100
015104  102401                                BVS     3$
015106  103001                                BCC     4$
                                        
015110  104001                        3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
                                        
015112  020412                        4$:     CMP     R4,(R2)         ;CORRECT RESULT?
015114  001401                                BEQ     5$              ;BR IF OK
                                        
015116  104001                                ERROR   1               ;SWAB DELIVERED WRONG RESULT
                                        
015120  013701  015140                5$:     MOV     @#20$,R1        ;LOAD R1 WITH TEST INSTR. WORD
015124  012702  063236                        MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
015130  012712  000377                        MOV     #377,(R2)       ;[DEST] = 000377
015134  000257                                CCC                     ;CLEAR FLAGS
015136  000273                                273                     ;MAKE N:C = 1011
```

```
4472
4473  015140  000322                  20$:    SWAB    (R2)+           ;TEST THE SWAB - DM2
4474
4475  015142  100403                          BMI     7$              ;N:C = 0100
4476  015144  001002                          BNE     7$
4477  015146  102401                          BVS     7$
4478  015150  103001                          BCC     6$
4479
4480  015152  104001                  7$:     ERROR   1               ;SWAB FAILED TO SET CODES PROPERLY
4481
4482  015154  020437  063236          6$:     CMP     R4,@#MBUF0      ;CORRECT RESULT ?
4483  015160  001401                          BEQ     8$              ;BR IF YES
4484
4485  015162  104001                  9$:     ERROR   1               ;SWAB DELIVERED THE WRONG RESULT
4486
4487  015164  020227  063240          8$:     CMP     R2,#MBUF0+2     ;DID AUTO INCREMENT OCCUR ?
4488  015170  001401                          BEQ     TST231          ;;BR IF YES
4489
4490  015172  104001                          ERROR   1               ;SWAB FAILED TO AUTO INC REG.
4491
4492                                  ;;*****************************************************************
4493                                  ;*TEST 231       SWAB MODE 1 TEST WITH NEG. RESULT
4494                                  ;;*****************************************************************
4495  015174                          TST231:
4496  015174  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4497  015176  012700  000231                  MOV     #231,R0         ;;LOAD R0 WITH TEST NUMBER
4498  015202  013701  015226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4499  015206  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
4500  015212  012702  063236                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4501  015216  012712  177400                  MOV     #177400,(R2)    ;SET UP DEST. OP = 177400
4502  015222  000257                          CCC                     ;CLEAR FLAGS
4503  015224  000267                          267                     ;MAKE N:C = 0111
4504
4505  015226  000312                  2$:     SWAB    (R2)            ;TEST THE SWAB
4506
4507  015230  100003                          BPL     3$
4508  015232  001402                          BEQ     3$              ;N:C = 1000?
4509  015234  102401                          BVS     3$
4510  015236  103001                          BCC     4$
4511
4512  015240  104001                  3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
4513
4514  015242  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT?
4515  015244  001401                          BEQ     TST232          ;;BR IF YES
4516
4517  015246  104001                  5$:     ERROR   1               ;SWAB DELIVERED WRONG RESULT
4518
4519                                  ;;*****************************************************************
4520                                  ;*TEST 232       NEG MODE 0 TEST : [DEST] = 0
4521                                  ;;*****************************************************************
4522  015250                          TST232:
4523  015250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4524  015252  012700  000232                  MOV     #232,R0         ;;LOAD R0 WITH TEST NUMBER
4525  015256  013701  015272                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4526  015262  005004                          CLR     R4              ;RESULT S / B = 0
4527  015264  005003                          CLR     R3              ;INITIAL [DEST] = 0
```

```
4528  015266  000257                          CCC                     ;CLEAR FLAGS
4529  015270  000273                          273                     ;MAKE N:C = 1011
4530
4531  015272  005403                  2$:     NEG     R3              ;TEST THE NEG
4532
4533  015274  100403                          BMI     3$
4534  015276  001002                          BNE     3$              ;N:C = 0100 ONLY "Z" SET?
4535  015300  102401                          BVS     3$
4536  015302  103001                          BCC     4$
4537
4538  015304  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
4539
4540  015306  020304                  4$:     CMP     R3,R4           ;WAS RESULT = 0
4541  015310  001401                          BEQ     TST233          ;;BR IF YES
4542
4543  015312  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
4544
4545                                  ;;*****************************************************************
4546                                  ;*TEST 233       NEG MODE 0 TEST : [DEST] LT 0
4547                                  ;;*****************************************************************
4548  015314                          TST233:
4549  015314  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4550  015316  012700  000233                  MOV     #233,R0         ;;LOAD R0 WITH TEST NUMBER
4551  015322  013701  015342                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4552  015326  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
4553  015332  012703  177776                  MOV     #-2,R3          ;INITIAL [DEST] = 177776
4554  015336  000257                          CCC                     ;CLEAR FLAGS
4555  015340  000276                          276                     ;MAKE N:C = 1110
4556
4557  015342  005403                  2$:     NEG     R3              ;TEST THE NEG
4558
4559  015344  100403                          BMI     3$
4560  015346  001402                          BEQ     3$              ;N:C = 0001?
4561  015350  102401                          BVS     3$
4562  015352  103401                          BCS     4$
4563
4564  015354  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
4565
4566  015356  020304                  4$:     CMP     R3,R4           ;RESULT = 2?
4567  015360  001401                          BEQ     TST234          ;;BR IF YES
4568
4569  015362  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
4570
4571                                  ;;*****************************************************************
4572                                  ;*TEST 234       NEG MODE 0 TEST : [DEST] = 100000 (8)
4573                                  ;;*****************************************************************
4574  015364                          TST234:
4575  015364  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4576  015366  012700  000234                  MOV     #234,R0         ;;LOAD R0 WITH TEST NUMBER
4577  015372  013701  015410                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4578  015376  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
4579  015402  010403                          MOV     R4,R3           ;INITIAL [DEST] = 100000
4580  015404  000257                          CCC                     ;CLEAR FLAGS
4581  015406  000264                          SEZ                     ;MAKE N:C = 01000
4582
4583  015410  005403                  2$:     NEG     R3              ;TEST THE NEG
```

```
4584
4585  015412  100003                          BPL     3$
4586  015414  001402                          BEQ     3$              ;N:C = 1011?
4587  015416  102001                          BVC     3$
4588  015420  103401                          BCS     4$
4589
4590  015422  104002                  3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
4591
4592  015424  020304                  4$:     CMP     R3,R4           ;RESULT STILL 100000?
4593  015426  001401                          BEQ     TST235          ;;BR IF YES
4594
4595  015430  104002                  5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
4596
4597                                  ;;**********************************************************************
4598                                  ;*TEST 235       NEG MODE 1 TEST : [DEST] = 0
4599                                  ;;**********************************************************************
4600  015432                          TST235:
4601  015432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4602  015434  012700  000235                  MOV     #235,R0         ;;LOAD R0 WITH TEST NUMBER
4603  015440  013701  015460                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4604  015444  012702  063236                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4605  015450  005004                          CLR     R4              ;RESULT S / B = 0
4606  015452  005012                          CLR     (R2)            ;INITIAL [DEST] = 0
4607  015454  000257                          CCC                     ;CLEAR FLAGS
4608  015456  000273                          273                     ;MAKE N:C = 1011
4609
4610  015460  005412                  2$:     NEG     (R2)            ;TEST THE NEG
4611
4612  015462  100403                          BMI     3$
4613  015464  001002                          BNE     3$              ;N:C = 0100?
4614  015466  102401                          BVS     3$
4615  015470  103001                          BCC     4$
4616
4617  015472  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
4618
4619  015474  021204                  4$:     CMP     (R2),R4         ;RESULT = 0?
4620  015476  001401                          BEQ     TST236          ;;BR IF YES
4621
4622  015500  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
4623
4624                                  ;;**********************************************************************
4625                                  ;*TEST 236       NEG MODE 1  TEST : [DEST] GT 0
4626                                  ;;**********************************************************************
4627  015502                          TST236:
4628  015502  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4629  015504  012700  000236                  MOV     #236,R0         ;;LOAD R0 WITH TEST NUMBER
4630  015510  013701  015534                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4631  015514  012702  063236                  MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4632  015520  012704  177776                  MOV     #-2,R4          ;RESULT S / B = 177776
4633  015524  012712  000002                  MOV     #2,(R2)         ;INITIAL [DEST] = 2
4634  015530  000257                          CCC                     ;CLEAR FLAGS
4635  015532  000266                          266                     ;MAKE N:C = 0110
4636
4637  015534  005412                  2$:     NEG     (2)             ;TEST THE NEG
4638
4639  015536  100003                          BPL     3$
```

```
4640  015540  001402                    BEQ     3$              ;N:C = 1001?
4641  015542  102401                    BVS     3$
4642  015544  103401                    BCS     4$
4643
4644  015546  104001            3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
4645
4646  015550  021204            4$:     CMP     (R2),R4         ;CORRECT RESULT?
4647  015552  001401                    BEQ     TST237          ;;BR IF YES
4648
4649  015554  104001            5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
4650
4651                            ;;**********************************************************************
4652                            ;*TEST 237      NEG MODE 1 TEST : [DEST] LT 0
4653                            ;;**********************************************************************
4654  015556                    TST237:
4655  015556  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4656  015560  012700  000237            MOV     #237,R0         ;;LOAD R0 WITH TEST NUMBER
4657  015564  013701  015610            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4658  015570  012702  063236            MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4659  015574  012704  000002            MOV     #2,R4           ;RESULT S / B = 2
4660  015600  012712  177776            MOV     #-2,(R2)        ;INITIAL [DEST] = 177776
4661  015604  000257                    CCC                     ;CLEAR FLAGS
4662  015606  000278                    278                     ;MAKE N:C = 1110
4663
4664  015610  005412            2$:     NEG     (R2)            ;TEST THE NEG
4665
4666  015612  100403                    BMI     3$
4667  015614  001402                    BEQ     3$              ;N:C = 0001?
4668  015616  102401                    BVS     3$
4669  015620  103401                    BCS     4$
4670
4671  015622  104001            3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
4672
4673  015624  021204            4$:     CMP     (R2),R4         ;CORRECT RESULT = 2?
4674  015626  001401                    BEQ     TST240          ;;BR IF YES
4675
4676  015630  104001            5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
4677
4678                            ;;**********************************************************************
4679                            ;*TEST 240      NEG MODE 1 TEST: [DEST] = 100000 (8)
4680                            ;;**********************************************************************
4681  015632                    TST240:
4682  015632  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4683  015634  012700  000240            MOV     #240,R0         ;;LOAD R0 WITH TEST NUMBER
4684  015640  013701  015662            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4685  015644  012702  063236            MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4686  015650  012704  100000            MOV     #100000,R4      ;RESULT S / B = 100000
4687  015654  010412                    MOV     R4,(R2)         ;INITIAL [DEST] = 100000
4688  015656  000257                    CCC                     ;CLEAR FLAGS
4689  015660  000264                    SEZ                     ;MAKE N:Z = 0100
4690
4691  015662  005412            2$:     NEG     (R2)            ;TEST THE NEG
4692
4693  015664  100003                    BPL     3$
4694  015666  001402                    BEQ     3$              ;N:C = 1011?
4695  015670  102001                    BVC     3$
```

```
4696  015672  103401                          BCS     4$
4697
4698  015674  104001                  3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
4699
4700  015676  021204                  4$:     CMP     (R2),R4         ;CORRECT RESULT = 100000?
4701  015700  001401                          BEQ     TST241          ;;BR IF YES
4702
4703  015702  104001                  5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
4704
4705                                  ;;**************************************************************
4706                                  ;*TEST 241       ROR TEST - DMO - N:C = 1110
4707                                  ;;**************************************************************
4708  015704                          TST241:
4709  015704  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4710  015706  012700  000241                  MOV     #241,R0         ;;LOAD R0 WITH TEST NUMBER
4711  015712  013701  015732                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4712  015716  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
4713  015722  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
4714  015726  000257                          CCC                     ;CLEAR FLAGS
4715  015730  000276                          276                     ;N:C = 1111
4716
4717  015732  006003                  2$:     ROR     R3              ;TEST THE ROR
4718
4719  015734  100403                          BMI     3$              ;N:C = 0000 ?
4720  015736  001402                          BEQ     3$
4721  015740  102401                          BVS     3$
4722  015742  103001                          BCC     4$
4723
4724  015744  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4725
4726  015746  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
4727  015750  001401                          BEQ     TST242          ;;BR IF YES
4728
4729  015752  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
4730
4731                                  ;;**************************************************************
4732                                  ;*TEST 242       ROR TEST - DMO - N:C = 1000
4733                                  ;;**************************************************************
4734  015754                          TST242:
4735  015754  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4736  015756  012700  000242                  MOV     #242,R0         ;;LOAD R0 WITH TEST NUMBER
4737  015762  013701  016000                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4738  015766  005004                          CLR     R4              ;RESULT S / B = 000000
4739  015770  012703  000001                  MOV     #1,R3           ;[DEST] = 1
4740  015774  000257                          CCC                     ;CLEAR FLAGS
4741  015776  000270                          SEN                     ;N:C = 1000
4742
4743  016000  006003                  2$:     ROR     R3              ;TEST THE ROR
4744
4745  016002  100403                          BMI     3$              ;N:C = 0111 ?
4746  016004  001002                          BNE     3$
4747  016006  102001                          BVC     3$
4748  016010  103401                          BCS     4$
4749
4750  016012  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4751
```

```
4752  016014  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
4753  016016  001401                          BEQ     TST243          ;;BR IF YES
4754
4755  016020  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
4756
4757                                  ;;**************************************************************
4758                                  ;*TEST 243       ROR TEST - DM0 - N:C = 0111
4759                                  ;;**************************************************************
4760  016022                          TST243:
4761  016022  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4762  016024  012700  000243                  MOV     #243,R0         ;;LOAD R0 WITH TEST NUMBER
4763  016030  013701  016050                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4764  016034  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
4765  016040  012703  052525                  MOV     #52525,R3       ;[DEST] = 052525
4766  016044  000257                          CCC                     ;CLEAR FLAGS
4767  016046  000267                          267                     ;N:C = 0111
4768
4769  016050  006003                  2$:     ROR     R3              ;TEST THE ROR
4770
4771  016052  100003                          BPL     3$              ;N:C = 1001 ?
4772  016054  001402                          BEQ     3$
4773  016056  102401                          BVS     3$
4774  016060  103401                          BCS     4$
4775
4776  016062  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4777
4778  016064  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
4779  016066  001401                          BEQ     TST244          ;;BR IF YES
4780
4781  016070  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
4782
4783                                  ;;**************************************************************
4784                                  ;*TEST 244       ASR TEST - DM0 - N:C = 1000
4785                                  ;;**************************************************************
4786  016072                          TST244:
4787  016072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4788  016074  012700  000244                  MOV     #244,R0         ;;LOAD R0 WITH TEST NUMBER
4789  016100  013701  016116                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4790  016104  005004                          CLR     R4              ;RESULT S / B = 000000
4791  016106  012703  000001                  MOV     #1,R3           ;[DEST] = 1
4792  016112  000257                          CCC                     ;CLEAR FLAGS
4793  016114  000270                          SEN                     ;N:C = 1000
4794
4795  016116  006003                  2$:     ROR     R3              ;TEST THE ROR
4796
4797  016120  100403                          BMI     3$              ;N:C = 0111 ?
4798  016122  001002                          BNE     3$
4799  016124  102001                          BVC     3$
4800  016126  103401                          BCS     4$
4801
4802  016130  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4803
4804  016132  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
4805  016134  001401                          BEQ     TST245          ;;BR IF YES
4806
4807  016136  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
```

```
4808
4809                                    ;;*****************************************************************
4810                                    ;*TEST 245      ASR TEST - DM0 - N:C = 0101
4811                                    ;;*****************************************************************
4812  016140                            TST245:
4813  016140  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4814  016142  012700  000245                   MOV     #245,R0         ;;LOAD R0 WITH TEST NUMBER
4815  016146  013701  016166                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4816  016152  012704  152525                   MOV     #152525,R4      ;RESULT S / B = 152525
4817  016156  012703  125252                   MOV     #125252,R3      ;[DEST] = 125252
4818  016162  000257                           CCC                     ;CLEAR FLAGS
4819  016164  000265                           265                     ;N:C = 0101
4820
4821  016166  006003                    2$:    ROR     R3              ;TEST THE ROR
4822
4823  016170  100003                           BPL     3$              ;N:C = 1010 ?
4824  016172  001402                           BEQ     3$
4825  016174  102001                           BVC     3$
4826  016176  103001                           BCC     4$
4827
4828  016200  104002                    3$:    ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4829
4830  016202  020403                    4$:    CMP     R4,R3           ;CORRECT RESULT ?
4831  016204  001401                           BEQ     TST246          ;;BR IF YES
4832
4833  016206  104002                    5$:    ERROR   2               ;ROR DELIVERED THE WRONG RESULT
4834
4835                                    ;;*****************************************************************
4836                                    ;*TEST 246      ASR TEST - DM0 - N:C = 1100
4837                                    ;;*****************************************************************
4838  016210                            TST246:
4839  016210  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4840  016212  012700  000246                   MOV     #246,R0         ;;LOAD R0 WITH TEST NUMBER
4841  016216  013701  016236                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4842  016222  012704  025252                   MOV     #25252,R4       ;RESULT S / B = 25252
4843  016226  012703  052525                   MOV     #52525,R3       ;[DEST] = 52525
4844  016232  000257                           CCC                     ;CLEAR FLAGS
4845  016234  000274                           274                     ;N:C = 1100
4846
4847  016236  006003                    2$:    ROR     R3              ;TEST THE ROR
4848
4849  016240  100403                           BMI     3$              ;N:C = 0011 ?
4850  016242  001402                           BEQ     3$
4851  016244  102001                           BVC     3$
4852  016246  103401                           BCS     4$
4853
4854  016250  104002                    3$:    ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
4855
4856  016252  020403                    4$:    CMP     R4,R3           ;CORRECT RESULT ?
4857  016254  001401                           BEQ     TST247          ;;BR IF YES
4858
4859  016256  104002                    5$:    ERROR   2               ;ROR DELIVERED THE WRONG RESULT
4860
4861                                    ;;*****************************************************************
4862                                    ;*TEST 247      ROR TEST - DM1 - N:C = 1110
4863                                    ;;*****************************************************************
```

```
4864  016260                          TST247:
4865  016260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4866  016262  012700  000247                  MOV     #247,R0         ;;LOAD R0 WITH TEST NUMBER
4867  016266  013701  016312                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4868  016272  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4869  016276  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
4870  016302  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
4871  016306  000257                          CCC                     ;CLEAR FLAGS
4872  016310  000276                          SEN                     ;N:C = 1110
4873
4874  016312  006012                  2$:     ROR     (R2)            ;TEST THE ROR
4875
4876  016314  100403                          BMI     3$              ;N:C = 0000 ?
4877  016316  001402                          BEQ     3$
4878  016320  102401                          BVS     3$
4879  016322  103001                          BCC     4$
4880
4881  016324  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
4882
4883  016326  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
4884  016330  001402                          BEQ     TST250          ;;BR IF YES
4885  016332  011203                          MOV     (R2),R3         ;GET THE WAS DATA
4886  016334  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
4887
4888                                  ;;**************************************************************
4889                                  ;*TEST 250      ROR TEST - DM1 - N:C = 1000
4890                                  ;;**************************************************************
4891  016336                          TST250:
4892  016336  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4893  016340  012700  000250                  MOV     #250,R0         ;;LOAD R0 WITH TEST NUMBER
4894  016344  013701  016366                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4895  016350  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4896  016354  005004                          CLR     R4              ;RESULT S / B = 000000
4897  016356  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
4898  016362  000257                          CCC                     ;CLEAR FLAGS
4899  016364  000270                          SEN                     ;N:C = 1000
4900
4901  016366  006012                  2$:     ROR     (R2)            ;TEST THE ROR
4902
4903  016370  100403                          BMI     3$              ;N:C = 0111 ?
4904  016372  001002                          BNE     3$
4905  016374  102001                          BVC     3$
4906  016376  103401                          BCS     4$
4907
4908  016400  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
4909
4910  016402  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
4911  016404  001402                          BEQ     TST251          ;;BR IF YES
4912
4913  016406  011203                          MOV     (R2),R3         ;GET THE WAS DATA
4914  016410  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
4915
4916                                  ;;**************************************************************
4917                                  ;*TEST 251      ROR TEST - DM1 - N:C = 0111
4918                                  ;;**************************************************************
4919  016412                          TST251:
```

```
4920  016412  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4921  016414  012700  000251                  MOV     #251,R0         ;;LOAD R0 WITH TEST NUMBER
4922  016420  013701  016444                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4923  016424  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4924  016430  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
4925  016434  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
4926  016440  000257                          CCC                     ;CLEAR FLAGS
4927  016442  000267                          267                     ;N:C = 0111
4928
4929  016444  006012              2$:         ROR     (R2)            ;TEST THE ROR
4930
4931  016446  100003                          BPL     3$              ;N:C = 1001 ?
4932  016450  001402                          BEQ     3$
4933  016452  102401                          BVS     3$
4934  016454  103401                          BCS     4$
4935
4936  016456  104001              3$:         ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
4937
4938  016460  020412              4$:         CMP     R4,(R2)         ;CORRECT RESULT ?
4939  016462  001402                          BEQ     TST252          ;;BR IF YES
4940
4941  016464  011203                          MOV     (R2),R3         ;GET THE WAS DATA
4942  016466  104001              5$:         ERROR   1               ;ROR DELIVERED WRONG RESULT
4943
4944                              ;;**************************************************************
4945                              ;*TEST 252      ASR TEST - DM1 - N:C = 1000
4946                              ;;**************************************************************
4947  016470                      TST252:
4948  016470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4949  016472  012700  000252                  MOV     #252,R0         ;;LOAD R0 WITH TEST NUMBER
4950  016476  013701  016520                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4951  016502  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4952  016506  005004                          CLR     R4              ;RESULT S  /  B = 000000
4953  016510  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
4954  016514  000257                          CCC                     ;CLEAR FLAGS
4955  016516  000270                          SEN                     ;N:C = 1000
4956
4957  016520  006012              2$:         ROR     (R2)            ;TEST THE ROR
4958
4959  016522  100403                          BMI     3$              ;N:C = 0111 ?
4960  016524  001002                          BNE     3$
4961  016526  102001                          BVC     3$
4962  016530  103401                          BCS     4$
4963
4964  016532  104001              3$:         ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
4965
4966  016534  020412              4$:         CMP     R4,(R2)         ;CORRECT RESULT ?
4967  016536  001402                          BEQ     TST253          ;;BR IF YES
4968
4969  016540  011203                          MOV     (R2),R3         ;GET THE WAS DATA
4970  016542  104001              5$:         ERROR   1               ;ROR DELIVERED WRONG RESULT
4971
4972                              ;;**************************************************************
4973                              ;*TEST 253      ASR TEST - DM1 - N:C = 1100
4974                              ;;**************************************************************
4975  016544                      TST253:
```

```
4976  016544  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4977  016546  012700  000253              MOV     #253,R0         ;;LOAD R0 WITH TEST NUMBER
4978  016552  013701  016576              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4979  016556  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
4980  016562  012704  025252              MOV     #25252,R4       ;RESULT S / B = 25252
4981  016566  012712  052525              MOV     #52525,(R2)     ;[DEST] = 52525
4982  016572  000257                      CCC                     ;CLEAR FLAGS
4983  016574  000274                      274                     ;N:C = 1100
4984
4985  016576  006012              2$:     ROR     (R2)            ;TEST THE ROR
4986
4987  016600  100403                      BMI     3$              ;N:C = 0011 ?
4988  016602  001402                      BEQ     3$
4989  016604  102001                      BVC     3$
4990  016606  103401                      BCS     4$
4991
4992  016610  104001              3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
4993
4994  016612  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
4995  016614  001402                      BEQ     TST254          ;;BR IF YES
4996
4997  016616  011203                      MOV     (R2),R3         ;GET THE WAS DATA
4998  016620  104001              5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
4999
5000                              ;;**************************************************************
5001                              ;*TEST 254      ASR TEST - DM1 - N:C = 0101
5002                              ;;**************************************************************
5003  016622                      TST254:
5004  016622  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5005  016624  012700  000254              MOV     #254,R0         ;;LOAD R0 WITH TEST NUMBER
5006  016630  013701  016654              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5007  016634  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5008  016640  012704  152525              MOV     #152525,R4      ;RESULT S / B = 152525
5009  016644  012712  125252              MOV     #125252,(R2)    ;[DEST] = 125252
5010  016650  000257                      CCC                     ;CLEAR FLAGS
5011  016652  000265                      265                     ;N:C = 0101
5012
5013  016654  006012              2$:     ROR     (R2)            ;TEST THE ROR
5014
5015  016656  100003                      BPL     3$              ;N:C = 1010 ?
5016  016660  001402                      BEQ     3$
5017  016662  102001                      BVC     3$
5018  016664  103001                      BCC     4$
5019
5020  016666  104001              3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5021
5022  016670  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5023  016672  001402                      BEQ     TST255          ;;BR IF YES
5024
5025  016674  011203                      MOV     (R2),R3         ;GET THE WAS DATA
5026  016676  104001              5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5027
5028                              ;;**************************************************************
5029                              ;*TEST 255      RORB TEST - DM2 - EVEN ADDRESS
5030                              ;;**************************************************************
5031  016700                      TST255:
```

```
5032  016700  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5033  016702  012700  000255                MOV     #255,R0         ;;LOAD R0 WITH TEST NUMBER
5034  016706  013701  016732                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5035  016712  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5036  016716  012704  000177                MOV     #177,R4         ;RESULT S / B = 177
5037  016722  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5038  016724  012712  000377                MOV     #377,(R2)       ;[DEST] = 377
5039  016730  000257                        CCC                     ;SCOPE SYNC "C" = 0
5040
5041  016732  106023              2$:       RORB    (R3)+           ;TEST THE RORB
5042
5043  016734  103401                        BCS     4$              ;BR IF ROR SET "C"
5044
5045  016736  104001              3$:       ERROR   1               ;ROR FAILED TO SET "C"
5046
5047  016740  022703  063237      4$:       CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5048  016744  001401                        BEQ     6$              ;BR IF YES
5049
5050  016746  104005              5$:       ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5051
5052  016750  020412              6$:       CMP     R4,(R2)         ;CORRECT RESULT ?
5053  016752  001402                        BEQ     TST256          ;;BR IF YES
5054
5055  016754  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5056  016756  104001              7$:       ERROR   1               ;RORB DELIVERED WRONG RESULT
5057
5058                              ;;**************************************************************
5059                              ;;TEST 256      RORB TEST - DM1 - EVEN ADDRESS
5060                              ;;**************************************************************
5061  016760                      TST256:
5062  016760  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5063  016762  012700  000256                MOV     #256,R0         ;;LOAD R0 WITH TEST NUMBER
5064  016766  013701  017014                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5065  016772  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5066  016776  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
5067  017002  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5068  017004  012712  000376                MOV     #376,(R2)       ;[DEST] = 376
5069  017010  000257                        CCC                     ;CLEAR FLAGS
5070  017012  000261                        SEC                     ;SCOPE SYNC - SET "C"
5071
5072  017014  106013              2$:       RORB    (R3)            ;TEST THE RORB
5073
5074  017016  103001                        BCC     4$              ;BR IF "C" CLR - IT SHOULD BE
5075
5076  017020  104001              3$:       ERROR   1               ;RORB FAILED TO CLR "C"
5077
5078  017022  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
5079  017024  001402                        BEQ     TST257          ;;BR IF YES
5080
5081  017026  011203                        MOV     (R2),R3         ;GET THE WAS DATA
5082  017030  104001              5$:       ERROR   1               ;RORB DELIVERED WRONG RESULT
5083
5084                              ;;**************************************************************
5085                              ;;TEST 257      RORB TEST - DM2 - ODD ADDRESS
5086                              ;;**************************************************************
5087  017032                      TST257:
```

```
5088  017032  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5089  017034  012700  000257                  MOV     #257,R0         ;;LOAD R0 WITH TEST NUMBER
5090  017040  013701  017102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5091                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT5
5092  017044  032737  000040  063160          BIT     #BIT5,@#BPTLOC  ;BREAKPOINT HALT SET ??
5093  017052  001401                          BEQ     .+4             ;BR IF NOT
5094  017054  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
5095  017056  012702  063237                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5096  017062  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
5097  017066  012705  063236                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5098  017072  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5099  017074  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
5100  017100  000257                          CCC                     ;SCOPE SYNC - "C" =0
5101
5102  017102  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5103
5104  017104  103401                          BCS     4$              ;BR IF "C" IS SET - IT SHOULD BE
5105
5106  017106  104001                  3$:     ERROR   1               ;RORB FAILED TO SET "C"
5107
5108  017110  022703  063240          4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5109  017114  001401                          BEQ     6$              ;BR IF YES
5110
5111  017116  104005                  5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5112
5113  017120  020415                  6$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5114  017122  001402                          BEQ     TST260          ;;BR IF YES
5115
5116  017124  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5117  017126  104001                  7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5118
5119                                  ;;**********************************************************************
5120                                  ;*TEST 260      RORB TEST - DM1 - ODD ADDRESS
5121                                  ;;**********************************************************************
5122  017130                          TST260:
5123  017130  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5124  017132  012700  000260                  MOV     #260,R0         ;;LOAD R0 WITH TEST NUMBER
5125  017136  013701  017166                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5126  017142  012702  063237                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5127  017146  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
5128  017152  012705  063236                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5129  017156  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5130  017160  012715  177377                  MOV     #177377,(R5)    ;[DEST] = 177377
5131  017164  000261                          SEC                     ;SCOPE SYNC - SET "C"
5132
5133  017166  106023                  2$:     RORB    (R3)+           ;TEST THE RORB
5134
5135  017170  103001                          BCC     4$              ;BR IF "C" CLEAR - IT SHOULD BE
5136
5137  017172  104001                  3$:     ERROR   1               ;RORB FAILED TO CLEAR "C"
5138
5139  017174  020415                  4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
5140  017176  001402                          BEQ     TST261          ;;BR IF YES
5141
5142  017200  011503                          MOV     (R5),R3         ;GET THE WAS DATA
5143  017202  104001                  5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
```

```
5144
5145                                    ;;*****************************************************************
5146                                    ;*TEST 261      ASRB TEST - DM2 - ODD ADDRESS
5147                                    ;;*****************************************************************
5148  017204                            TST261:
5149  017204  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5150  017206  012700  000261                   MOV     #261,R0         ;;LOAD R0 WITH TEST NUMBER
5151  017212  013701  017242                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5152  017216  012702  063237                   MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5153  017222  012704  000377                   MOV     #377,R4         ;RESULT S / B = 377
5154  017226  012705  063236                   MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5155  017232  010203                           MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5156  017234  012715  000777                   MOV     #777,(R5)       ;[DEST] = 777
5157  017240  000257                           CCC                     ;SCOPE SYNC "C" = 0
5158
5159  017242  106223                    2$:    ASRB    (R3)+           ;TEST THE ASRB
5160
5161  017244  103401                           BCS     4$              ;BR IF CARRY SET - IT SHOULD BE
5162
5163  017246  104001                    3$:    ERROR   1               ;ASRB FAILED TO SET THE CARRY
5164
5165  017250  022703  063240           4$:    CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
5166  017254  001401                           BEQ     6$              ;BR IF YES
5167
5168  017256  104005                    5$:    ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5169
5170  017260  020415                    6$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5171  017262  001402                           BEQ     TST262          ;;BR IF YES
5172
5173  017264  011503                           MOV     (R5),R3         ;GET THE WAS DATA
5174  017266  104001                    7$:    ERROR   1               ;ASRB DELIVERED WRONG RESULT
5175
5176                                    ;;*****************************************************************
5177                                    ;*TEST 262      ASRB TEST - DM1 - ODD ADDRESS
5178                                    ;;*****************************************************************
5179  017270                            TST262:
5180  017270  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5181  017272  012700  000262                   MOV     #262,R0         ;;LOAD R0 WITH TEST NUMBER
5182  017276  013701  017326                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5183  017302  012702  063237                   MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
5184  017306  012704  140377                   MOV     #140377,R4      ;RESULT S / B = 140377
5185  017312  012705  063236                   MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
5186  017316  010203                           MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5187  017320  012715  100377                   MOV     #100377,(R5)    ;[DEST] = 100377
5188  017324  000261                           SEC                     ;SCOPE SYNC - "C" = 1
5189
5190  017326  106213                    2$:    ASRB    (R3)            ·TEST THE ASRB
5191
5192  017330  103001                           BCC     4$              ;BR IF CARRY CLEAR - IT SHOULD BE
5193
5194  017332  104001                    3$:    ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5195
5196  017334  020415                    4$:    CMP     R4,(R5)         ;CORRECT RESULT ?
5197  017336  001402                           BEQ     TST263          ;;BR IF YES
5198
5199  017340  011503                           MOV     (R5),R3         ;GET THE WAS DATA
```

```
5200  017342  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5201
5202                                  ;;*****************************************************************
5203                                  ;*TEST 263       ASRB TEST - DM2 - EVEN ADDRESS
5204                                  ;;*****************************************************************
5205  017344                          TST263:
5206  017344  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5207  017346  012700  000263                  MOV     #263,R0         ;;LOAD R0 WITH TEST NUMBER
5208  017352  013701  017376                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5209  017356  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5210  017362  012704  000077                  MOV     #77,R4          ;RESULT S / B = 77
5211  017366  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5212  017370  012712  000177                  MOV     #177,(R2)       ;[DEST] = 177
5213  017374  000257                          CCC                     ;SCOPE SYNC - "C" = 0
5214
5215  017376  106223                  2$:     ASRB    (R3)+           ;TEST THE ASRB
5216
5217  017400  103401                          BCS     4$              ;BR IF "C" = 1 - IT SHOULD BE
5218
5219  017402  104001                  3$:     ERROR   1               ;ASRB FAILED TO SET "C"
5220
5221  017404  022703  063237          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5222  017410  001401                          BEQ     6$              ;BR IF YES
5223
5224  017412  104005                  5$:     ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
5225
5226  017414  020412                  6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5227  017416  001402                          BEQ     TST264          ;;BR IF YES
5228
5229  017420  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5230  017422  104001                  7$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
5231
5232                                  ;;*****************************************************************
5233                                  ;*TEST 264       ASRB TEST - DM1 - EVEN ADDRESS
5234                                  ;;*****************************************************************
5235  017424                          TST264:
5236  017424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5237  017426  012700  000264                  MOV     #264,R0         ;;LOAD R0 WITH TEST NUMBER
5238  017432  013701  017456                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5239  017436  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5240  017442  012704  000303                  MOV     #303,R4         ;RESULT S / B = 303
5241  017446  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5242  017450  012712  000206                  MOV     #206,(R2)       ;[DEST] = 206
5243  017454  000261                          SEC                     ;SCOPE SYNC - "C" = 1
5244
5245  017456  106213                  2$:     ASRB    (R3)            ;TEST THE CLRASRB
5246
5247  017460  103001                          BCC     4$              ;BR IF CARRY CLEAR - ITSHOULD BE
5248
5249  017462  104001                  3$:     ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
5250
5251  017464  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5252  017466  001402                          BEQ     TST265          ;;BR IF YES
5253
5254  017470  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5255  017472  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
```

```
5256
5257                                    ;;*****************************************************************
5258                                    ;*TEST 265      TST DMO TEST - N:C = 1011
5259                                    ;;*****************************************************************
5260  017474                            TST265:
5261  017474  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5262  017476  012700  000265                    MOV     #265,R0         ;;LOAD R0 WITH TEST NUMBER
5263  017502  013701  017516                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5264  017506  005004                            CLR     R4              ;RESULT S / B = 000000
5265  017510  005003                            CLR     R3              ;[DEST] = 000000
5266  017512  000257                            CCC                     ;CLEAR CODES
5267  017514  000273                            273                     ;N:C=1011
5268
5269  017516  005703                    2$:     TST     R3              ;TEST THE TST
5270
5271  017520  100403                            BMI     3$              ;N:C = 0100 ?
5272  017522  001002                            BNE     3$
5273  017524  102401                            BVS     3$
5274  017526  103001                            BCC     4$
5275
5276  017530  104002                    3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY
5277
5278  017532  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5279  017534  001401                            BEQ     TST266          ;;BR IF YES
5280
5281  017536  104002                    5$:     ERROR   2               ;TST ALTERED THE [DEST]
5282
5283                                    ;;*****************************************************************
5284                                    ;*TEST 266      TST DMO TEST - N:C = 0100
5285                                    ;;*****************************************************************
5286  017540                            TST266:
5287  017540  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5288  017542  012700  000266                    MOV     #266,R0         ;;LOAD R0 WITH TEST NUMBER
5289  017546  013701  017564                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5290  017552  005004                            CLR     R4
5291  017554  005104                            COM     R4              ;RESULT S / B = 177777
5292  017556  010403                            MOV     R4,R3           ;[DEST] = 177777
5293  017560  000257                            CCC                     ;CLEAR CODES
5294  017562  000264                            264                     ;N:C=0100
5295
5296  017564  005703                    2$:     TST     R3              ;TEST THE TST
5297
5298  017566  100003                            BPL     3$              ;N:C = 1000 ?
5299  017570  001402                            BEQ     3$
5300  017572  102401                            BVS     3$
5301  017574  103001                            BCC     4$
5302
5303  017576  104002                    3$:     ERROR   2               ;TST FAILED TO ALTER CODES PROPERLY
5304
5305  017600  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5306  017602  001401                            BEQ     TST267          ;;BR IF YES
5307
5308  017604  104002                    5$:     ERROR   2               ;TST ALTERED THE [DEST]
5309
5310
5311                                    ;;*****************************************************************
                                        ;*TEST 267      CLR DMO TEST - N:C = 1011
```

```
5312                                    ;;*****************************************************************
5313  017606                            TST267:
5314  017606  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5315  017610  012700  000267                    MOV     #267,R0         ;;LOAD R0 WITH TEST NUMBER
5316  017614  013701  017632                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5317  017620  005004                            CLR     R4              ;RESULT S / B = 000000
5318  017622  012703  177777                    MOV     #-1,R3          ;[DEST] = 177777
5319  017626  000257                            CCC                     ;CLEAR CODES
5320  017630  000273                            273                     ;N:C = 1011
5321
5322  017632  005003                    2$:     CLR     R3              ;TEST THE CLR
5323
5324  017634  100403                            BMI     3$              ;N:C = 0100 ?
5325  017636  001002                            BNE     3$
5326  017640  102401                            BVS     3$
5327  017642  103001                            BCC     4$
5328
5329  017644  104002                    3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
5330
5331  017646  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5332  017650  001401                            BEQ     TST270          ;;BR IF YES
5333
5334  017652  104002                    5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
5335
5336                                    ;;*****************************************************************
5337                                    ;TEST 270       CLR DMO TEST - N:C = 0000
5338                                    ;;*****************************************************************
5339  017654                            TST270:
5340  017654  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5341  017656  012700  000270                    MOV     #270,R0         ;;LOAD R0 WITH TEST NUMBER
5342  017662  013701  017676                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5343  017666  005004                            CLR     R4              ;RESULT S / B = 000000
5344  017670  012703  177777                    MOV     #-1,R3          ;[DEST] = 177777
5345  017674  000257                            CCC                     ;CLEAR CODES
5346
5347  017676  005003                    2$:     CLR     R3              ;TEST THE CLR
5348
5349  017700  100403                            BMI     3$              ;N:C = 0100 ?
5350  017702  001002                            BNE     3$
5351  017704  102401                            BVS     3$
5352  017706  103001                            BCC     4$
5353
5354  017710  104002                    3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY
5355
5356  017712  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5357  017714  001401                            BEQ     TST271          ;;BR IF YES
5358
5359  017716  104002                    5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT
5360
5361                                    ;;*****************************************************************
5362                                    ;TEST 271       COM DMO TEST - N:C = 0110
5363                                    ;;*****************************************************************
5364  017720                            TST271:
5365  017720  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5366  017722  012700  000271                    MOV     #271,R0         ;;LOAD R0 WITH TEST NUMBER
5367  017726  013701  017746                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
5368  017732  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
5369  017736  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
5370  017742  000257                  CCC                     ;CLEAR CODES
5371  017744  000266                  266                     ;N:C = 0110
5372
5373  017746  005103          2$:     COM     R3              ;TEST THE COM
5374
5375  017750  100003                  BPL     3$              ;N:C = 1001 ?
5376  017752  001402                  BEQ     3$
5377  017754  102401                  BVS     3$
5378  017756  103401                  BCS     4$
5379
5380  017760  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
5381
5382  017762  020403          4$:     CMP     R4,R3           ;RESULT OK ?
5383  017764  001401                  BEQ     TST272          ;;BR IF YES
5384
5385  017766  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
5386
5387                          ;;**************************************************************
5388                          ;*TEST 272       COM DMO TEST - N:C = 1001
5389                          ;;**************************************************************
5390  017770                  TST272:
5391  017770  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5392  017772  012700  000272          MOV     #272,R0         ;;LOAD R0 WITH TEST NUMBER
5393  017776  013701  020014          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5394  020002  005004                  CLR     R4              ;RESULT S / B = 000000
5395  020004  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
5396  020010  000257                  CCC                     ;CLEAR CODES
5397  020012  000271                  271                     ;N:C = 1001
5398
5399  020014  005103          2$:     COM     R3              ;TEST THE COM
5400
5401  020016  100403                  BMI     3$              ;N:C = 0101 ?
5402  020020  001002                  BNE     3$
5403  020022  102401                  BVS     3$
5404  020024  103401                  BCS     4$
5405
5406  020026  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
5407
5408  020030  020403          4$:     CMP     R4,R3           ;RESULT OK ?
5409  020032  001401                  BEQ     TST273          ;;BR IF YES
5410
5411  020034  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
5412
5413                          ;;**************************************************************
5414                          ;*TEST 273       INC DMO TEST - N:C = 1011
5415                          ;;**************************************************************
5416  020036                  TST273:
5417  020036  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5418  020040  012700  000273          MOV     #273,R0         ;;LOAD R0 WITH TEST NUMBER
5419  020044  013701  020062          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5420  020050  005004                  CLR     R4              ;RESULT S / B = 000000
5421  020052  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
5422  020056  000257                  CCC                     ;CLEAR CODES
5423  020060  000273                  273                     ;N:C = 1011
```

```
5424
5425  020062  005203                  2$:     INC     R3              ;TEST THE INC
5426
5427  020064  100403                          BMI     3$              ;N:C = 0101 ?
5428  020066  001002                          BNE     3$
5429  020070  102401                          BVS     3$
5430  020072  103401                          BCS     4$
5431
5432  020074  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
5433
5434  020076  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5435  020100  001401                          BEQ     TST274          ;;BR IF YES
5436
5437  020102  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
5438
5439                                  ;;**********************************************************************
5440                                  ;*TEST 274       INC DMO TEST - N:C = 0100
5441                                  ;;**********************************************************************
5442  020104                          TST274:
5443  020104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5444  020106  012700  000274                  MOV     #274,R0         ;;LOAD R0 WITH TEST NUMBER
5445  020112  013701  020132                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5446  020116  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
5447  020122  012703  077777                  MOV     #77777,R3       ;[DEST] = 77777
5448  020126  000257                          CCC                     ;CLEAR CODES
5449  020130  000264                          264                     ;N:C = 0100
5450
5451  020132  005203                  2$:     INC     R3              ;TEST THE INC
5452
5453  020134  100003                          BPL     3$              ;N:C = 1010 ?
5454  020136  001402                          BEQ     3$
5455  020140  102001                          BVC     3$
5456  020142  103001                          BCC     4$
5457
5458  020144  104002                  3$:     ERROR   2               ;INC FAILED TO ALTER THE CODES PROPERLY
5459
5460  020146  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5461  020150  001401                          BEQ     TST275          ;;BR IF YES
5462
5463  020152  104002                  5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT
5464
5465                                  ;;**********************************************************************
5466                                  ;*TEST 275       DEC DMO TEST - N:C = 1011
5467                                  ;;**********************************************************************
5468  020154                          TST275:
5469  020154  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5470  020156  012700  000275                  MOV     #275,R0         ;;LOAD R0 WITH TEST NUMBER
5471  020162  013701  020200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5472  020166  005004                          CLR     R4              ;RESULT S / B = 000000
5473  020170  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5474  020174  000257                          CCC                     ;CLEAR CODES
5475  020176  000273                          273                     ;N:C = 1011
5476
5477  020200  005303                  2$:     DEC     R3              ;TEST THE DEC
5478
5479  020202  100403                          BMI     3$              ;N:C = 0101 ?
```

```
5480  020204  001002                        BNE     3$
5481  020206  102401                        BVS     3$
5482  020210  103401                        BCS     4$
5483
5484  020212  104002              3$:       ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
5485
5486  020214  020403              4$:       CMP     R4,R3           ;RESULT OK ?
5487  020216  001401                        BEQ     TST276          ;;BR IF YES
5488
5489  020220  104002              5$:       ERROR   2               ;DEC DELIVERED THE WRONG RESUL
5490
5491                                        ;;*****************************************************************
5492                                        ;*TEST 276      DEC DMO TEST - N:C = 1100
5493                                        ;;*****************************************************************
5494  020222                                TST276:
5495  020222  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5496  020224  012700  000276                MOV     #276,R0         ;;LOAD R0 WITH TEST NUMBER
5497  020230  013701  020250                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5498  020234  012704  077777                MOV     #77777,R4       ;RESULT S / B = 77777
5499  020240  012703  100000                MOV     #100000,R3      ;[DEST] = 100000
5500  020244  000257                        CCC                     ;CLEAR CODES
5501  020246  000274                        274                     ;N:C = 1100
5502
5503  020250  005303              2$:       DEC     R3              ;TEST THE DEC
5504
5505  020252  100403                        BMI     3$              ;N:C = 0010 ?
5506  020254  001402                        BEQ     3$
5507  020256  102001                        BVC     3$
5508  020260  103001                        BCC     4$
5509
5510  020262  104002              3$:       ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
5511
5512  020264  020403              4$:       CMP     R4,R3           ;RESULT OK ?
5513  020266  001401                        BEQ     TST277          ;;BR IF YES
5514
5515  020270  104002              5$:       ERROR   2               ;DEC DELIVERED THE WRONG RESULT
5516
5517                                        ;;*****************************************************************
5518                                        ;*TEST 277      DEC DMO TEST - N:C = 0000
5519                                        ;;*****************************************************************
5520  020272                                TST277:
5521  020272  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5522  020274  012700  000277                MOV     #277,R0         ;;LOAD R0 WITH TEST NUMBER
5523  020300  013701  020314                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5524  020304  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
5525  020310  005003                        CLR     R3              ;[DEST] = 000000
5526  020312  000257                        CCC                     ;CLEAR CODES
5527
5528  020314  005303              2$:       DEC     R3              ;TEST THE DEC
5529
5530  020316  100003                        BPL     3$              ;N:C = 1000 ?
5531  020320  001402                        BEQ     3$
5532  020322  102401                        BVS     3$
5533  020324  103001                        BCC     4$
5534
5535  020326  104002              3$:       ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
```

```
5536
5537  020330  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5538  020332  001401                          BEQ     TST300          ;;BR IF YES
5539
5540  020334  104002                  5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
5541
5542                                  ;;**********************************************************************
5543                                  ;*TEST 300       ASL DMO TEST - N:C = 1000
5544                                  ;;**********************************************************************
5545  020336                          TST300:
5546  020336  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5547  020340  012700  000300                  MOV     #300,R0         ;;LOAD R0 WITH TEST NUMBER
5548  020344  013701  020362                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5549  020350  005004                          CLR     R4              ;RESULT S / B = 000000
5550  020352  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
5551  020356  000257                          CCC                     ;CLEAR CODES
5552  020360  000270                          SEN                     ;N:C = 1000
5553
5554  020362  006303                  2$:     ASL     R3              ;TEST THE ASL
5555
5556  020364  100403                          BMI     3$              ;N:C = 0111 ?
5557  020366  001002                          BNE     3$
5558  020370  102001                          BVC     3$
5559  020372  103401                          BCS     4$
5560
5561  020374  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
5562
5563  020376  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5564  020400  001401                          BEQ     TST301          ;;BR IF YES
5565
5566  020402  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
5567
5568                                  ;;**********************************************************************
5569                                  ;*TEST 301       ASL DMO TEST - N:C = 0101
5570                                  ;;**********************************************************************
5571  020404                          TST301:
5572  020404  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5573  020406  012700  000301                  MOV     #301,R0         ;;LOAD R0 WITH TEST NUMBER
5574  020412  013701  020432                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5575  020416  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
5576  020422  012703  040000                  MOV     #40000,R3       ;[DEST] = 40000
5577  020426  000257                          CCC                     ;CLEAR CODES
5578  020430  000265                          265                     ;N:C = 0101
5579
5580  020432  006303                  2$:     ASL     R3              ;TEST THE ASL
5581
5582  020434  100003                          BPL     3$              ;N:C = 1010 ?
5583  020436  001402                          BEQ     3$
5584  020440  102001                          BVC     3$
5585  020442  103001                          BCC     4$
5586
5587  020444  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
5588
5589  020446  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5590  020450  001401                          BEQ     TST302          ;;BR IF YES
5591
```

```
5592  020452  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
5593
5594                                  ;;*****************************************************************
5595                                  ;*TEST 302       ASL DMO TEST - N:C = 0010
5596                                  ;;*****************************************************************
5597  020454                          TST302:
5598  020454  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5599  020456  012700  000302                  MOV     #302,R0         ;;LOAD R0 WITH TEST NUMBER
5600  020462  013701  020476                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5601  020466  005004                          CLR     R4              ;RESULT S  /  B = 000000
5602  020470  005003                          CLR     R3              ;[DEST] = 000000
5603  020472  000257                          CCC                     ;CLEAR CODES
5604  020474  000262                          SEV                     ;N:C = 0010
5605
5606  020476  006303                  2$:     ASL     R3              ;TEST THE ASL
5607
5608  020500  100403                          BMI     3$              ;N:C = 0100 ?
5609  020502  001002                          BNE     3$
5610  020504  102401                          BVS     3$
5611  020506  103001                          BCC     4$
5612
5613  020510  104002                  3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
5614
5615  020512  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5616  020514  001401                          BEQ     TST303          ;;BR IF YES
5617
5618  020516  104002                  5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
5619
5620                                  ;;*****************************************************************
5621                                  ;*TEST 303       ROL DMO TEST - N:C = 1101
5622                                  ;;*****************************************************************
5623  020520                          TST303:
5624  020520  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5625  020522  012700  000303                  MOV     #303,R0         ;;LOAD R0 WITH TEST NUMBER
5626  020526  013701  020546                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5627  020532  012704  052525                  MOV     #52525,R4       ;RESULT S  /  B = 52525
5628  020536  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
5629  020542  000257                          CCC                     ;CLEAR CODES
5630  020544  000275                          275                     ;N:C = 1101
5631
5632  020546  006103                  2$:     ROL     R3              ;TEST THE ROL
5633
5634  020550  100403                          BMI     3$              ;N:C = 0011 ?
5635  020552  001402                          BEQ     3$
5636  020554  102001                          BVC     3$
5637  020556  103401                          BCS     4$
5638
5639  020560  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
5640
5641  020562  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5642  020564  001401                          BEQ     TST304          ;;BR IF YES
5643
5644  020566  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
5645
5646                                  ;;*****************************************************************
5647                                  ;*TEST 304       ROL DMO TEST - N:C = 0101
```

```
5648                                    ;;*****************************************************************
5649  020570                            ;TST304:
5650  020570  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5651  020572  012700  000304                    MOV     #304,R0         ;;LOAD R0 WITH TEST NUMBER
5652  020576  013701  020616                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5653  020602  012704  125253                    MOV     #125253,R4      ;RESULT S / B = 125253
5654  020606  012703  052525                    MOV     #52525,R3       ;[DEST] = 52525
5655  020612  000257                            CCC                     ;CLEAR CODES
5656  020614  000265                            265                     ;N:C = 0101
5657
5658  020616  006103                    2$:     ROL     R3              ;TEST THE ROL
5659
5660  020620  100003                            BPL     3$              ;N:C = 1010 ?
5661  020622  001402                            BEQ     3$
5662  020624  102001                            BVC     3$
5663  020626  103001                            BCC     4$
5664
5665  020630  104002                    3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
5666  020632  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5667  020634  001401                            BEQ     TST305          ;;BR IF YES
5668
5669  020636  104002                    5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
5670
5671                                    ;;*****************************************************************
5672                                    ;*TEST 305      ROL DMO TEST - N:C = 0010
5673                                    ;*****************************************************************
5674  020640                            TST305:
5675  020640  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5676  020642  012700  000305                    MOV     #305,R0         ;;LOAD R0 WITH TEST NUMBER
5677  020646  013701  020662                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5678  020652  005004                            CLR     R4              ;RESULT S / B = 000000
5679  020654  005003                            CLR     R3              ;[DEST] = 000000
5680  020656  000257                            CCC                     ;CLEAR CODES
5681  020660  000262                            SEV                     ;N:C = 0010
5682
5683  020662  006103                    2$:     ROL     R3              ;TEST THE ROL
5684
5685  020664  100403                            BMI     3$              ;N:C = 0100 ?
5686  020666  001002                            BNE     3$
5687  020670  102401                            BVS     3$
5688  020672  103001                            BCC     4$
5689
5690  020674  104002                    3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
5691
5692  020676  020403                    4$:     CMP     R4,R3           ;RESULT OK ?
5693  020700  001401                            BEQ     TST306          ;;BR IF YES
5694
5695  020702  104002                    5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
5696
5697                                    ;;*****************************************************************
5698                                    ;*TEST 306      ADC DMO TEST - N:C = 0101
5699                                    ;*****************************************************************
5700  020704                            TST306:
5701  020704  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5702  020706  012700  000306                    MOV     #306,R0         ;;LOAD R0 WITH TEST NUMBER
5703  020712  013701  020732                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
5704  020716  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
5705  020722  012703  077777          MOV     #77777,R3       ;[DEST] = 77777
5706  020726  000257                  CCC                     ;CLEAR CODES
5707  020730  000265                  265                     ;N:C = 0101
5708
5709  020732  005503          2$:     ADC     R3              ;TEST THE ADC
5710
5711  020734  100003                  BPL     3$              ;N:C = 1010 ?
5712  020736  001402                  BEQ     3$
5713  020740  102001                  BVC     3$
5714  020742  103001                  BCC     4$
5715
5716  020744  104002          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
5717
5718  020746  020403          4$:     CMP     R4,R3           ;RESULT OK ?
5719  020750  001401                  BEQ     TST307          ;;BR IF YES
5720
5721  020752  104002          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
5722
5723                          ;;*****************************************************************
5724                          ;*TEST 307       ADC DMO TEST - N:C = 1011
5725                          ;;*****************************************************************
5726  020754                  TST307:
5727  020754  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5728  020756  012700  000307          MOV     #307,R0         ;;LOAD R0 WITH TEST NUMBER
5729  020762  013701  021000          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5730  020766  005004                  CLR     R4              ;RESULT S / B = 000000
5731  020770  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
5732  020774  000257                  CCC                     ;CLEAR CODES
5733  020776  000273                  273                     ;N:C = 1011
5734
5735  021000  005503          2$:     ADC     R3              ;TEST THE ADC
5736
5737  021002  100403                  BMI     3$              ;N:C = 0101 ?
5738  021004  001002                  BNE     3$
5739  021006  102401                  BVS     3$
5740  021010  103401                  BCS     4$
5741
5742  021012  104002          3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
5743
5744  021014  020403          4$:     CMP     R4,R3           ;RESULT OK ?
5745  021016  001401                  BEQ     TST310          ;;BR IF YES
5746
5747  021020  104002          5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
5748
5749                          ;;*****************************************************************
5750                          ;*TEST 310       ADC DMO TEST - N:C = 1010
5751                          ;;*****************************************************************
5752  021022                  TST310:
5753  021022  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5754  021024  012700  000310          MOV     #310,R0         ;;LOAD R0 WITH TEST NUMBER
5755  021030  013701  021050          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5756  021034  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
5757  021040  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
5758  021044  000257                  CCC                     ;CLEAR CODES
5759  021046  000272                  272                     ;N:C = 1010
```

```
5760
5761  021050  005503                  2$:     ADC     R3              ;TEST THE ADC
5762
5763  021052  100003                          BPL     3$              ;N:C = 1000 ?
5764  021054  001402                          BEQ     3$
5765  021056  102401                          BVS     3$
5766  021060  103001                          BCC     4$
5767
5768  021062  104002                  3$:     ERROR   2               ;ADC FAILED TO ALTER THE CODES PROPERLY
5769
5770  021064  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5771  021066  001401                          BEQ     TST311          ;;BR IF YES
5772
5773  021070  104002                  5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT
5774
5775                                  ;;**************************************************************
5776                                  ;*TEST 311       SBC DMO TEST - N:C = 1011
5777                                  ;;**************************************************************
5778  021072                          TST311:
5779  021072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5780  021074  012700  000311                  MOV     #311,R0         ;;LOAD R0 WITH TEST NUMBER
5781  021100  013701  021116                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5782  021104  005004                          CLR     R4              ;RESULT S / B = 000000
5783  021106  012703  000001                  MOV     #1,R3           ;[DEST] = +1
5784  021112  000257                          CCC                     ;CLEAR CODES
5785  021114  000273                          273                     ;N:C = 1011
5786
5787  021116  005603                  2$:     SBC     R3              ;TEST THE SBC
5788
5789  021120  100403                          BMI     3$              ;N:C = 0100 ?
5790  021122  001002                          BNE     3$
5791  021124  102401                          BVS     3$
5792  021126  103001                          BCC     4$
5793
5794  021130  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
5795
5796  021132  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5797  021134  001401                          BEQ     TST312          ;;BR IF YES
5798
5799  021136  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
5800
5801                                  ;;**************************************************************
5802                                  ;*TEST 312       SBC DMO TEST - N:C = 0101
5803                                  ;;**************************************************************
5804  021140                          TST312:
5805  021140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5806  021142  012700  000312                  MOV     #312,R0         ;;LOAD R0 WITH TEST NUMBER
5807  021146  013701  021166                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5808  021152  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
5809  021156  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
5810  021162  000257                          CCC                     ;CLEAR CODES
5811  021164  000265                          265                     ;N:C = 0101
5812
5813  021166  005603                  2$:     SBC     R3              ;TEST THE SBC
5814
5815  021170  100403                          BMI     3$              ;N:C = 0010 ?
```

```
5816  021172  001402                          BEQ     3$
5817  021174  102001                          BVC     3$
5818  021176  103001                          BCC     4$
5819
5820  021200  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
5821
5822  021202  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5823  021204  001401                          BEQ     TST313          ;;BR IF YES
5824
5825  021206  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
5826
5827                                  ;;**************************************************************
5828                                  ;*TEST 313       SBC DMO TEST - N:C = 1110
5829                                  ;;**************************************************************
5830  021210                          TST313:
5831  021210  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5832  021212  012700  000313                  MOV     #313,R0         ;;LOAD R0 WITH TEST NUMBER
5833  021216  013701  021236                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5834  021222  012704  000001                  MOV     #1,R4           ;RESULT S / B = 1
5835  021226  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5836  021232  000257                          CCC                     ;CLEAR CODES
5837  021234  000276                          276                     ;N:C = 1110
5838
5839  021236  005603                  2$:     SBC     R3              ;TEST THE SBC
5840
5841  021240  100403                          BMI     3$              ;N:C = 0000 ?
5842  021242  001402                          BEQ     3$
5843  021244  102401                          BVS     3$
5844  021246  103001                          BCC     4$
5845
5846  021250  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
5847
5848  021252  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5849  021254  001401                          BEQ     TST314          ;;BR IF YES
5850
5851  021256  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
5852
5853                                  ;;**************************************************************
5854                                  ;*TEST 314       SBC DMO TEST - N:C = 0111
5855                                  ;;**************************************************************
5856  021260                          TST314:
5857  021260  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5858  021262  012700  000314                  MOV     #314,R0         ;;LOAD R0 WITH TEST NUMBER
5859  021266  013701  021304                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5860  021272  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
5861  021276  005003                          CLR     R3              ;[DEST] = 000000
5862  021300  000257                          CCC                     ;CLEAR CODES
5863  021302  000267                          267                     ;N:C = 0111
5864
5865  021304  005603                  2$:     SBC     R3              ;TEST THE SBC
5866
5867  021306  100003                          BPL     3$              ;N:C = 1001 ?
5868  021310  001402                          BEQ     3$
5869  021312  102401                          BVS     3$
5870  021314  103401                          BCS     4$
5871
```

```
5872  021316  104002                  3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
5873
5874  021320  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
5875  021322  001401                          BEQ     TST315          ;;BR IF YES
5876
5877  021324  104002                  5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
5878
5879                                  ;;*********************************************************************
5880                                  ;*TEST 315      TST DM1 TEST - N:C = 1011
5881                                  ;;*********************************************************************
5882  021326                          TST315:
5883  021326  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5884  021330  012700  000315                  MOV     #315,R0         ;;LOAD R0 WITH TEST NUMBER
5885  021334  013701  021354                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5886  021340  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5887  021344  005004                          CLR     R4              ;RESULT S / B = 000000
5888  021346  005012                          CLR     (R2)            ;[DEST] = 000000
5889  021350  000257                          CCC                     ;CLEAR CODES
5890  021352  000273                          273                     ;N:C=1011
5891
5892  021354  005712                  2$:     TST     (R2)            ;TEST THE TST
5893
5894  021356  100403                          BMI     3$              ;N:C = 0100 ?
5895  021360  001002                          BNE     3$
5896  021362  102401                          BVS     3$
5897  021364  103001                          BCC     4$
5898
5899  021366  104001                  3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
5900
5901  021370  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
5902  021372  001402                          BEQ     TST316          ;;BR IF YES
5903
5904  021374  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5905  021376  104001                  5$:     ERROR   1               ;TST ALTERED THE [DEST]
5906
5907                                  ;;*********************************************************************
5908                                  ;*TEST 316      TST DM1 TEST - N:C = 0100
5909                                  ;;*********************************************************************
5910  021400                          TST316:
5911  021400  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5912  021402  012700  000316                  MOV     #316,R0         ;;LOAD R0 WITH TEST NUMBER
5913  021406  013701  021432                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5914  021412  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5915  021416  005004                          CLR     R4
5916  021420  005104                          COM     R4              ;RESULT S / B = 177777
5917  021422  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
5918  021426  000257                          CCC                     ;CLEAR CODES
5919  021430  000264                          264                     ;N:C=0100
5920
5921  021432  005712                  2$:     TST     (R2)            ;TEST THE TST
5922
5923  021434  100003                          BPL     3$              ;N:C = 1000 ?
5924  021436  001402                          BEQ     3$
5925  021440  102401                          BVS     3$
5926  021442  103001                          BCC     4$
5927
```

```
5928  021444  104001                  3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
5929
5930  021446  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
5931  021450  001402                          BEQ     TST317          ;;BR IF YES
5932
5933  021452  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5934  021454  104001                  5$:     ERROR   1               ;TST ALTERED THE [DEST]
5935
5936                                  ;;**************************************************************
5937                                  ;*TEST 317      CLR DM1 TEST - N:C = 1011
5938                                  ;;**************************************************************
5939  021456                          TST317:
5940  021456  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5941  021460  012700  000317                  MOV     #317,R0         ;;LOAD R0 WITH TEST NUMBER
5942  021464  013701  021520                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5943                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT6
5944  021470  032737  000100  063160          BIT     #BIT6,@#BPTLOC  ;BREAKPOINT HALT SET ??
5945  021476  001401                          BEQ     .+4             ;BR IF NOT
5946  021500  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
5947  021502  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5948  021506  005004                          CLR     R4              ;RESULT S / B = 000000
5949  021510  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
5950  021514  000257                          CCC                     ;CLEAR CODES
5951  021516  000273                          273                     ;N:C = 1011
5952
5953  021520  005012                  2$:     CLR     (R2)            ;TEST THE CLR
5954
5955  021522  100403                          BMI     3$              ;N:C = 0100 ?
5956  021524  001002                          BNE     3$
5957  021526  102401                          BVS     3$
5958  021530  103001                          BCC     4$
5959
5960  021532  104001                  3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
5961
5962  021534  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
5963  021536  001402                          BEQ     TST320          ;;BR IF YES
5964
5965  021540  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5966  021542  104001                  5$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
5967
5968                                  ;;**************************************************************
5969                                  ;*TEST 320      CLR DM2 TEST - N:C = 0000
5970                                  ;;**************************************************************
5971  021544                          TST320:
5972  021544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5973  021546  012700  000320                  MOV     #320,R0         ;;LOAD R0 WITH TEST NUMBER
5974  021552  013701  021572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5975  021556  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5976  021562  005004                          CLR     R4              ;RESULT S / B = 000000
5977  021564  013712  063250                  MOV     @#DWTA+2,(R2)   ;[DEST] = 177777
5978  021570  000257                          CCC                     ;CLEAR CODES
5979
5980  021572  005022                  2$:     CLR     (R2)+           ;TEST THE CLR
5981
5982  021574  100403                          BMI     3$              ;N:C = 0100 ?
5983  021576  001002                          BNE     3$
```

```
5984  021600  102401                  BVS     3$
5985  021602  103001                  BCC     4$
5986
5987  021604  104001          3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
5988
5989  021606  022702  063240  4$:     CMP     #MBUF0+2,R2     ;DID CLR INCREMENT DEST REG
5990  021612  001401                  BEQ     6$              ;BR IF YES
5991
5992  021614  104005          5$:     ERROR   5               ;CLR FAILED TO UPDATE DEST REG
5993
5994  021616  020442          6$:     CMP     R4,-(R2)                ;RESULT OK ?
5995  021620  001402                  BEQ     TST321          ;;BR IF YES
5996
5997  021622  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5998  021624  104001          7$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
5999
6000                          ;;**************************************************************
6001                          ;*TEST 321      COM DM1 TEST - N:C = 0110
6002                          ;;**************************************************************
6003  021626                  TST321:
6004  021626  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6005  021630  012700  000321          MOV     #321,R0         ;;LOAD R0 WITH TEST NUMBER
6006  021634  013701  021660          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6007  021640  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6008  021644  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6009  021650  012712  052525          MOV     #52525,(R2)     ;(DEST) = 52525
6010  021654  000257                  CCC                     ;CLEAR CODES
6011  021656  000266                  266                     ;N:C = 0110
6012
6013  021660  005112          2$:     COM     (R2)            ;TEST THE CLR
6014
6015  021662  100003                  BPL     3$              ;N:C = 1001 ?
6016  021664  001402                  BEQ     3$
6017  021666  102401                  BVS     3$
6018  021670  103401                  BCS     4$
6019
6020  021672  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6021  021674  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6022  021676  001402                  BEQ     TST322          ;;BR IF YES
6023
6024  021700  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6025  021702  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6026
6027                          ;;**************************************************************
6028                          ;*TEST 322      COM DM1 TEST - N:C = 1001
6029                          ;;**************************************************************
6030  021704                  TST322:
6031  021704  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6032  021706  012700  000322          MOV     #322,R0         ;;LOAD R0 WITH TEST NUMBER
6033  021712  013701  021734          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6034  021716  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6035  021722  005004                  CLR     R4              ;RESULT S / B = 000000
6036  021724  012712  177777          MOV     #-1,(R2)        ;(DEST) = 177777
6037  021730  000257                  CCC                     ;CLEAR CODES
6038  021732  000271                  271                     ;N:C = 1001
6039
```

```
6040  021734  005112                  2$:     COM     (R2)            ;TEST THE COM
6041
6042  021736  100403                          BMI     3$              ;N:C = 0101 ?
6043  021740  001002                          BNE     3$
6044  021742  102401                          BVS     3$
6045  021744  103401                          BCS     4$
6046
6047  021746  104001                  3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6048  021750  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6049  021752  001402                          BEQ     TST323          ;;BR IF YES
6050
6051  021754  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6052  021756  104001                  5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6053
6054                                  ;;**************************************************************
6055                                  ;*TEST 323       INC DM1 TEST - N:C = 1011
6056                                  ;;**************************************************************
6057  021760                          TST323:
6058  021760  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6059  021762  012700  000323                  MOV     #323,R0         ;;LOAD R0 WITH TEST NUMBER
6060  021766  013701  022010                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6061  021772  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6062  021776  005004                          CLR     R4              ;RESULT S / B = 000000
6063  022000  012712  177777                  MOV     #-1,(R2)                ;[DEST] = 177777
6064  022004  000257                          CCC                     ;CLEAR CODES
6065  022006  000273                          273                     ;N:C = 1011
6066
6067  022010  005212                  2$:     INC     (R2)            ;TEST THE INC
6068
6069  022012  100403                          BMI     3$              ;N:C = 0101 ?
6070  022014  001002                          BNE     3$
6071  022016  102401                          BVS     3$
6072  022020  103401                          BCS     4$
6073
6074  022022  104001                  3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6075  022024  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6076  022026  001402                          BEQ     TST324          ;;BR IF YES
6077
6078  022030  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6079  022032  104001                  5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6080
6081                                  ;;**************************************************************
6082                                  ;*TEST 324       INC DM1 TEST - N:C = 0100
6083                                  ;;**************************************************************
6084  022034                          TST324:
6085  022034  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6086  022036  012700  000324                  MOV     #324,R0         ;;LOAD R0 WITH TEST NUMBER
6087  022042  013701  022066                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WOFO
6088  022046  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6089  022052  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6090  022056  012712  077777                  MOV     #77777,(R2)     ;[DEST] = 77777
6091  022062  000257                          CCC                     ;CLEAR CODES
6092  022064  000264                          264                     ;N:C = 0100
6093
6094  022066  005212                  2$:     INC     (R2)            ;TEST THE INC
6095
```

```
6096  022070  100003                          BPL     3$              ;N:C = 1010 ?
6097  022072  001402                          BEQ     3$
6098  022074  102001                          BVC     3$
6099  022076  103001                          BCC     4$
6100
6101  022100  104001                  3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
6102  022102  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6103  022104  001402                          BEQ     TST325          ;;BR IF YES
6104
6105  022106  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6106  022110  104001                  5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
6107
6108                                  ;;**********************************************************************
6109                                  ;*TEST 325       DEC DM1 TEST - N:C = 1011
6110                                  ;;**********************************************************************
6111  022112                          TST325:
6112  022112  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6113  022114  012700  000325                  MOV     #325,R0         ;;LOAD R0 WITH TEST NUMBER
6114  022120  013701  022142                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6115  022124  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6116  022130  005004                          CLR     R4              ;RESULT S / B = 000000
6117  022132  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
6118  022136  000257                          CCC                     ;CLEAR CODES
6119  022140  000273                          273                     ;N:C = 1011
6120
6121  022142  005312                  2$:     DEC     (R2)            ;TEST THE DEC
6122
6123  022144  100403                          BMI     3$              ;N:C = 0101 ?
6124  022146  001002                          BNE     3$
6125  022150  102401                          BVS     3$
6126  022152  103401                          BCS     4$
6127
6128  022154  104001                  3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6129  022156  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6130  022160  001402                          BEQ     TST326          ;;BR IF YES
6131
6132  022162  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6133  022164  104001                  5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6134
6135                                  ;;**********************************************************************
6136                                  ;*TEST 326       DEC DM1 TEST - N:C = 1100
6137                                  ;;**********************************************************************
6138  022166                          TST326:
6139  022166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6140  022170  012700  000326                  MOV     #326,R0         ;;LOAD R0 WITH TEST NUMBER
6141  022174  013701  022220                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6142  022200  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6143  022204  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
6144  022210  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
6145  022214  000257                          CCC                     ;CLEAR CODES
6146  022216  000274                          274                     ;N:C = 1100
6147
6148  022220  005312                  2$:     DEC     (R2)            ;TEST THE DEC
6149
6150  022222  100403                          BMI     3$              ;N:C = 0010 ?
6151  022224  001402                          BEQ     3$
```

```
6152  022226  102001                  BVC     3$
6153  022230  103001                  BCC     4$
6154
6155  022232  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6156  022234  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6157  022236  001402                  BEQ     TS†327          ;;BR IF YES
6158
6159  022240  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6160  022242  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6161
6162                          ;;**************************************************************
6163                          ;*TEST 327      DEC DM1 TEST - N:C = 0000
6164                          ;;**************************************************************
6165  022244                  †ST327:
6166  022244  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6167  022246  012700  000327          MOV     #327,R0         ;;LOAD R0 WITH TEST NUMBER
6168  022252  013701  022272          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6169  022256  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6170  022262  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
6171  022266  005012                  CLR     (R2)            ;[DEST] = 000000
6172  022270  000257                  CCC                     ;CLEAR CODES
6173
6174  022272  005312          2$:     DEC     (R2)            ;TEST THE DEC
6175
6176  022274  100003                  BPL     3$              ;N:C = 1000 ?
6177  022276  001402                  BEQ     3$
6178  022300  102401                  BVS     3$
6179  022302  103001                  BCC     4$
6180
6181  022304  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
6182  022306  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6183  022310  001402                  BEQ     TS†330          ;;BR IF YES
6184
6185  022312  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6186  022314  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
6187
6188                          ;;**************************************************************
6189                          ;*TEST 330      ASL DM1 TEST - N:C = 1000
6190                          ;;**************************************************************
6191  022316                  †ST330:
6192  022316  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6193  022320  012700  000330          MOV     #330,R0         ;;LOAD R0 WITH TEST NUMBER
6194  022324  013701  022346          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6195  022330  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6196  022334  005004                  CLR     R4              ;RESULT S / B = 000000
6197  022336  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
6198  022342  000257                  CCC                     ;CLEAR CODES
6199  022344  000270                  SEN                     ;N:C = 1000
6200
6201  022346  006312          2$:     ASL     (R2)            ;TEST THE ASL
6202
6203  022350  100403                  BMI     3$              ;N:C = 0111 ?
6204  022352  001002                  BNE     3$
6205  022354  102001                  BVC     3$
6206  022356  103401                  BCS     4$
6207
```

```
6208  022360  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6209  022362  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6210  022364  001402                          BEQ     TST331          ;;BR IF YES
6211
6212  022366  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6213  022370  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
6214
6215                                  ;;**********************************************************************
6216                                  ;*TEST 331       ASL DM1 TEST - N:C = 0101
6217                                  ;;**********************************************************************
6218  022372                          TST331:
6219  022372  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6220  022374  012700  000331                  MOV     #331,R0         ;;LOAD R0 WITH TEST NUMBER
6221  022400  013701  022424                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6222  022404  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6223  022410  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6224  022414  012712  040000                  MOV     #40000,(R2)     ;[DEST] = 40000
6225  022420  000257                          CCC                     ;CLEAR CODES
6226  022422  000265                          265                     ;N:C = 0101
6227
6228  022424  006312                  2$:     ASL     (R2)            ;TEST THE ASL
6229
6230  022426  100003                          BPL     3$              ;N:C = 1010 ?
6231  022430  001402                          BEQ     3$
6232  022432  102001                          BVC     3$
6233  022434  103001                          BCC     4$
6234
6235  022436  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6236  022440  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6237  022442  001402                          BEQ     TST332          ;;BR IF YES
6238
6239  022444  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6240  022446  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
6241
6242                                  ;;**********************************************************************
6243                                  ;*TEST 332       ASL DM1 TEST - N:C = 0010
6244                                  ;;**********************************************************************
6245  022450                          TST332:
6246  022450  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6247  022452  012700  000332                  MOV     #332,R0         ;;LOAD R0 WITH TEST NUMBER
6248  022456  013701  022476                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6249  022462  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6250  022466  005004                          CLR     R4              ;RESULT S / B = 000000
6251  022470  005012                          CLR     (R2)            ;[DEST] = 000000
6252  022472  000257                          CCC                     ;CLEAR CODES
6253  022474  000262                          SEV                     ;N:C = 0010
6254
6255  022476  006312                  2$:     ASL     (R2)            ;TEST THE ASL
6256
6257  022500  100403                          BMI     3$              ;N:C = 0100 ?
6258  022502  001002                          BNE     3$
6259  022504  102401                          BVS     3$
6260  022506  103001                          BCC     4$
6261
6262  022510  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
6263  022512  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
```

```
006264  022514  001402                          BEQ     TST333          ;;BR IF YES
006265
006266  022516  011203                          MOV     (R2),R3         ;GET THE WAS DATA
006267  022520  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
006268
006269                                  ;;**********************************************************************
006270                                  ;*TEST 333      ROL DM1 TEST - N:C = 1101
006271                                  ;;**********************************************************************
006272  022522                          TST333:
006273  022522  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
006274  022524  012700  000333                  MOV     #333,R0         ;;LOAD R0 WITH TEST NUMBER
006275  022530  013701  022554                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
006276  022534  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006277  022540  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
006278  022544  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
006279  022550  000257                          CCC                     ;CLEAR CODES
006280  022552  000275                          275                     ;N:C = 1101
006281
006282  022554  006112                  2$:     ROL     (R2)            ;TEST THE ROL
006283
006284  022556  100403                          BMI     3$              ;N:C = 0011 ?
006285  022560  001402                          BEQ     3$
006286  022562  102001                          BVC     3$
006287  022564  103401                          BCS     4$
006288
006289  022566  104001                  3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
006290  022570  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
006291  022572  001402                          BEQ     TST334          ;;BR IF YES
006292
006293  022574  011203                          MOV     (R2),R3         ;GET THE WAS DATA
006294  022576  104001                  5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
006295
006296                                  ;;**********************************************************************
006297                                  ;*TEST 334      ROL DM1 TEST - N:C = 0101
006298                                  ;;**********************************************************************
006299  022600                          TST334:
006300  022600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
006301  022602  012700  000334                  MOV     #334,R0         ;;LOAD R0 WITH TEST NUMBER
006302  022606  013701  022632                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
006303  022612  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
006304  022616  012704  125253                  MOV     #125253,R4      ;RESULT S / B = 125253
006305  022622  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
006306  022626  000257                          CCC                     ;CLEAR CODES
006307  022630  000265                          265                     ;N:C = 0101
006308
006309  022632  006112                  2$:     ROL     (R2)            ;TEST THE ROL
006310
006311  022634  100003                          BPL     3$              ;N:C = 1010 ?
006312  022636  001402                          BEQ     3$
006313  022640  102001                          BVC     3$
006314  022642  103001                          BCC     4$
006315
006316  022644  104001                  3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
006317  022646  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
006318  022650  001402                          BEQ     TST335          ;;BR IF YES
006319
```

```
6320  022652  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6321  022654  104001                  5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
6322
6323                                  ;;**********************************************************************
6324                                  ;*TEST 335       ROL DM1 TEST - N:C = 0010
6325                                  ;;**********************************************************************
6326  022656                          TST335:
6327  022656  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6328  022660  012700  000335                  MOV     #335,R0         ;;LOAD R0 WITH TEST NUMBER
6329  022664  013701  022704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6330  022670  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6331  022674  005004                          CLR     R4              ;RESULT S / B = 000000
6332  022676  005012                          CLR     (R2)            ;[DEST] = 000000
6333  022700  000257                          CCC                     ;CLEAR CODES
6334  022702  000262                          SEV                     ;N:C = 0010
6335
6336  022704  006112                  2$:     ROL     (R2)            ;TEST THE ROL
6337
6338  022706  100403                          BMI     3$              ;N:C = 0100 ?
6339  022710  001002                          BNE     3$
6340  022712  102401                          BVS     3$
6341  022714  103001                          BCC     4$
6342
6343  022716  104001                  3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
6344  022720  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6345  022722  001402                          BEQ     TST336          ;;BR IF YES
6346
6347  022724  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6348  022726  104001                  5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
6349
6350                                  ;;**********************************************************************
6351                                  ;*TEST 336       ADC DM1 TEST - N:C = 0101
6352                                  ;;**********************************************************************
6353  022730                          TST336:
6354  022730  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6355  022732  012700  000336                  MOV     #336,R0         ;;LOAD R0 WITH TEST NUMBER
6356  022736  013701  022762                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6357  022742  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6358  022746  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
6359  022752  012712  077777                  MOV     #77777,(R2)     ;[DEST] = 77777
6360  022756  000257                          CCC                     ;CLEAR CODES
6361  022760  000265                          265                     ;N:C = 0101
6362
6363  022762  005512                  2$:     ADC     (R2)            ;TEST THE ADC
6364
6365  022764  100003                          BPL     3$              ;N:C = 1010 ?
6366  022766  001402                          BEQ     3$
6367  022770  102001                          BVC     3$
6368  022772  103001                          BCC     4$
6369
6370  022774  104001                  3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
6371  022776  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
6372  023000  001402                          BEQ     TST337          ;;BR IF YES
6373
6374  023002  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6375  023004  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
```

```
6376
6377                                    ;;*****************************************************************
6378                                    ;*TEST 337      ADC DM1 TEST - N:C = 1011
6379                                    ;;*****************************************************************
6380  023006                            TST337:
6381  023006  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6382  023010  012700  000337                   MOV     #337,R0         ;;LOAD R0 WITH TEST NUMBER
6383  023014  013701  023036                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6384  023020  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6385  023024  005004                           CLR     R4              ;RESULT S  /  B = 000000
6386  023026  012712  177777                   MOV     #-1,(R2)        ;[DEST] = 177777
6387  023032  000257                           CCC                     ;CLEAR CODES
6388  023034  000273                           273                     ;N:C = 1011
6389
6390  023036  005512                    2$:    ADC     (R2)            ;TEST THE ADC
6391
6392  023040  100403                           BMI     3$              ;N:C = 0101 ?
6393  023042  001002                           BNE     3$
6394  023044  102401                           BVS     3$
6395  023046  103401                           BCS     4$
6396
6397  023050  104001                    3$:    ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
6398  023052  020412                    4$:    CMP     R4,(R2)         ;RESULT OK ?
6399  023054  001402                           BEQ     TST340          ;;BR IF YES
6400
6401  023056  011203                           MOV     (R2),R3         ;GET THE WAS DATA
6402  023060  104001                    5$:    ERROR   1               ;ADC DELIVERED THE WRONG RESULT
6403
6404                                    ;;*****************************************************************
6405                                    ;*TEST 340      ADC DM1 TEST - N:C = 1010
6406                                    ;;*****************************************************************
6407  023062                            TST340:
6408  023062  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6409  023064  012700  000340                   MOV     #340,R0         ;;LOAD R0 WITH TEST NUMBER
6410  023070  013701  023114                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6411  023074  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6412  023100  012704  177777                   MOV     #-1,R4          ;RESULT S  /  B = 177777
6413  023104  012712  177777                   MOV     #-1,(R2)        ;[DEST] = 177777
6414  023110  000257                           CCC                     ;CLEAR CODES
6415  023112  000272                           272                     ;N:C = 1010
6416
6417  023114  005512                    2$:    ADC     (R2)            ;TEST THE ADC
6418
6419  023116  100003                           BPL     3$              ;N:C = 1000 ?
6420  023120  001402                           BEQ     3$
6421  023122  102401                           BVS     3$
6422  023124  103001                           BCC     4$
6423
6424  023126  104001                    3$:    ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
6425  023130  020412                    4$:    CMP     R4,(R2)         ;RESULT OK ?
6426  023132  001402                           BEQ     TST341          ;;BR IF YES
6427
6428  023134  011203                           MOV     (R2),R3         ;GET THE WAS DATA
6429  023136  104001                    5$:    ERROR   1               ;ADC DELIVERED THE WRONG RESULT
6430
6431                                    ;;*****************************************************************
```

```
                                ;*TEST 341      SBC DM1 TEST - N:C = 1011
                                ;;*****************************************************************
023140                          TST341:
023140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023142  012700  000341                  MOV     #341,R0         ;;LOAD R0 WITH TEST NUMBER
023146  013701  023170                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023152  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023156  005004                          CLR     R4              ;RESULT S / B = 000000
023160  012712  000001                  MOV     #1,(R2)         ;[DEST] = +1
023164  000257                          CCC                     ;CLEAR CODES
023166  000273                          273                     ;N:C = 1011

023170  005612                  2$:     SBC     (R2)            ;TEST THE SBC

023172  100403                          BMI     3$              ;N:C = 0100 ?
023174  001002                          BNE     3$
023176  102401                          BVS     3$
023200  103001                          BCC     4$

023202  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023204  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023206  001402                          BEQ     TST342          ;;BR IF YES

023210  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023212  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 342      SBC DM1 TEST - N:C = 0101
                                ;;*****************************************************************
023214                          TST342:
023214  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023216  012700  000342                  MOV     #342,R0         ;;LOAD R0 WITH TEST NUMBER
023222  013701  023246                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023226  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023232  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
023236  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
023242  000257                          CCC                     ;CLEAR CODES
023244  000265                          265                     ;N:C = 0101

023246  005612                  2$:     SBC     (R2)            ;TEST THE SBC

023250  100403                          BMI     3$              ;N:C = 0010 ?
023252  001402                          BEQ     3$
023254  102001                          BVC     3$
023256  103001                          BCC     4$

023260  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023262  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023264  001402                          BEQ     TST343          ;;BR IF YES

023266  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023270  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 343      SBC DM1 TEST - N:C = 1110
                                ;;*****************************************************************
```

```
                                TST343:
023272                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023272  000004                          MOV     #343,R0         ;;LOAD R0 WITH TEST NUMBER
023274  012700  000343                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023300  013701  023324                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023304  012702  063236                  MOV     #1,R4           ;RESULT S / B = 1
023310  012704  000001                  MOV     #1,(R2)         ;[DEST] = 1
023314  012712  000001                  CCC                     ;CLEAR CODES
023320  000257                          276                     ;N:C = 1110
023322  000276
023324  005612                  2$:     SBC     (R2)            ;TEST THE SBC
023326  100403                          BMI     3$              ;N:C = 0000 ?
023330  001402                          BEQ     3$
023332  102401                          BVS     3$
023334  103001                          BCC     4$
023336  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023340  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023342  001402                          BEQ     TST344          ;;BR IF YES
023344  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023346  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT

                                ;;**************************************************************
                                ;*TEST 344      SBC DM1 TEST - N:C = 0111
                                ;;**************************************************************
023350                          TST344:
023350  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023352  012700  000344                  MOV     #344,R0         ;;LOAD R0 WITH TEST NUMBER
023356  013701  023400                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023362  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023366  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
023372  005012                          CLR     (R2)            ;[DEST] = 000000
023374  000257                          CCC                     ;CLEAR CODES
023376  000267                          267                     ;N:C = 0111
023400  005612                  2$:     SBC     (R2)            ;TEST THE SBC
023402  100003                          BPL     3$              ;N:C = 1001 ?
023404  001402                          BEQ     3$
023406  102401                          BVS     3$
023410  103401                          BCS     4$
023412  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023414  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
023416  001402                          BEQ     TST345          ;;BR IF YES
023420  011203                          MOV     (R2),R3         ;GET THE WAS DATA
023422  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT

                                ;;**************************************************************
                                ;*TEST 345      NEGB - MODE 0 TEST - N:C = 0110
                                ;;**************************************************************
023424                          TST345:
023424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
6544  023426  012700  000345          MOV     #345,R0         ;;LOAD R0 WITH TEST NUMBER
6545  023432  013701  023452          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6546  023436  012704  177776          MOV     #177776,R4      ;RESULT S / B = 376 (LO BYTE)
6547  023442  012703  177402          MOV     #177402,R3      ;[DEST] = 177402
6548  023446  000257                  CCC                     ;CLEAR FLAGS
6549  023450  000266                  266                     ;N:C = 0110
6550
6551  023452  105403          2$:     NEGB    R3              ;TEST THE NEGB
6552
6553  023454  100003                  BPL     3$              ;N:C = 1001
6554  023456  001402                  BEQ     3$
6555  023460  102401                  BVS     3$
6556  023462  103401                  BCS     4$
6557
6558  023464  104002          3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
6559
6560  023466  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
6561  023470  001401                  BEQ     TST346          ;;BR IF YES
6562
6563  023472  104002          5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
6564
6565                          ;;**************************************************************
6566                          ;*TEST 346       NEGB - MODE 0 TEST - N:C = 0011
6567                          ;;**************************************************************
6568  023474                  TST346:
6569  023474  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6570  023476  012700  000346          MOV     #346,R0         ;;LOAD R0 WITH TEST NUMBER
6571  023502  013701  023522          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6572  023506  012704  177400          MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
6573  023512  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
6574  023516  000257                  CCC                     ;CLEAR FLAGS
6575  023520  000263                  263                     ;N:C = 0011
6576
6577  023522  105403          2$:     NEGB    R3              ;TEST THE NEGB
6578
6579  023524  100403                  BMI     3$              ;N:C = 0100
6580  023526  001002                  BNE     3$
6581  023530  102401                  BVS     3$
6582  023532  103001                  BCC     4$
6583
6584  023534  104002          3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
6585
6586  023536  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
6587  023540  001401                  BEQ     TST347          ;;BR IF YES
6588
6589  023542  104002          5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
6590
6591                          ;;**************************************************************
6592                          ;*TEST 347       NEGB - MODE 0 TEST - N:C = 1101
6593                          ;;**************************************************************
6594  023544                  TST347:
6595  023544  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6596  023546  012700  000347          MOV     #347,R0         ;;LOAD R0 WITH TEST NUMBER
6597  023552  013701  023572          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6598  023556  012704  177600          MOV     #177600,R4      ;RESULT S / B = 200 (LO BYTE)
6599  023562  012703  177600          MOV     #177600,R3      ;[DEST] = 177600
```

```
6600  023566  000257                          CCC                     ;CLEAR FLAGS
6601  023570  000275                          275                     ;N:C = 1101
6602
6603  023572  105403                  2$:     NEGB    R3              ;TEST THE NEGB
6604
6605  023574  100003                          BPL     3$              ;N:C = 1011
6606  023576  001402                          BEQ     3$
6607  023600  102001                          BVC     3$
6608  023602  103401                          BCS     4$
6609
6610  023604  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
6611
6612  023606  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
6613  023610  001401                          BEQ     TST350          ;;BR IF YES
6614
6615  023612  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
6616
6617                                  ;;**************************************************************
6618                                  ;*TEST 350       CLRB - MODE 0 TEST - N:C = 1011
6619                                  ;;**************************************************************
6620  023614                          TST350:
6621  023614  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6622  023616  012700  000350                  MOV     #350,R0         ;;LOAD R0 WITH TEST NUMBER
6623  023622  013701  023642                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6624  023626  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
6625  023632  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
6626  023636  000257                          CCC                     ;CLEAR FLAGS
6627  023640  000273                          273                     ;N:C = 1011
6628
6629  023642  105003                  2$:     CLRB    R3              ;TEST THE CLRB
6630
6631  023644  100403                          BMI     3$              ;N:C = 0100 ?
6632  023646  001002                          BNE     3$
6633  023650  102401                          BVS     3$
6634  023652  103001                          BCC     4$
6635
6636  023654  104002                  3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
6637
6638  023656  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
6639  023660  001401                          BEQ     TST351          ;;BR IF YES
6640
6641  023662  104002                  5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
6642
6643                                  ;;**************************************************************
6644                                  ;*TEST 351       CLRB - MODE 0 TEST - N:C = 0100
6645                                  ;;**************************************************************
6646  023664                          TST351:
6647  023664  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6648  023666  012700  000351                  MOV     #351,R0         ;;LOAD R0 WITH TEST NUMBER
6649  023672  013701  023712                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6650  023676  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
6651  023702  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
6652  023706  000257                          CCC                     ;CLEAR FLAGS
6653  023710  000264                          SEZ                     ;N:C = 0100
6654
6655  023712  105003                  2$:     CLRB    R3              ;TEST THE CLRB
```

```
6656
6657  023714  100403                          BMI     3$              ;N:C = 0100 ?
6658  023716  001002                          BNE     3$
6659  023720  102401                          BVS     3$
6660  023722  103001                          BCC     4$
6661
6662  023724  104002                  3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
6663
6664  023726  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
6665  023730  001401                          BEQ     TST352          ;;BR IF YES
6666
6667  023732  104002                  5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
6668
6669                                  ;;**************************************************************
6670                                  ;*TEST 352       CLRB TEST - DM2 - ODD ADDRESS
6671                                  ;;**************************************************************
6672  023734                          TST352:
6673  023734  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6674  023736  012700  000352                  MOV     #352,R0         ;;LOAD R0 WITH TEST NUMBER
6675  023742  013701  023772                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6676  023746  012702  063237                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
6677  023752  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
6678  023756  012705  063236                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
6679  023762  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6680  023764  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
6681  023770  000257                          CCC                     ;SCOPE SYNC
6682
6683  023772  105023                  2$:     CLRB    (R3)+           ;TEST THE CLRB
6684
6685  023774  022703  063240                  CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
6686  024000  001401                          BEQ     4$              ;BR IF YES
6687
6688  024002  104005                  3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
6689
6690  024004  020415                  4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
6691  024006  001402                          BEQ     TST353          ;;BR IF YES
6692
6693  024010  011503                          MOV     (R5),R3         ;GET THE WAS DATA
6694  024012  104001                  5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
6695
6696                                  ;;**************************************************************
6697                                  ;*TEST 353       CLRB TEST - DM1 - ODD ADDRESS
6698                                  ;;**************************************************************
6699  024014                          TST353:
6700  024014  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6701  024016  012700  000353                  MOV     #353,R0         ;;LOAD R0 WITH TEST NUMBER
6702  024022  013701  024052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6703  024026  012702  063237                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
6704  024032  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
6705  024036  012705  063236                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
6706  024042  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6707  024044  012715  177777                  MOV     #-1,(R5)        ;[DEST] = 177777
6708  024050  000257                          CCC                     ;SCOPE SYNC
6709
6710  024052  105013                  2$:     CLRB    (R3)            ;TEST THE CLRB
6711
```

```
6712  024054  020415                          CMP     R4,(R5)         ;CORRECT RESULT ?
6713  024056  001402                          BEQ     TST354          ;;BR IF YES
6714
6715  024060  011503                          MOV     (R5),R3         ;GET THE WAS DATA
6716  024062  104001                  3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
6717
6718                                  ;;**********************************************************************
6719                                  ;*TEST 354       CLRB TEST - DM2 - EVEN ADDRESS
6720                                  ;;**********************************************************************
6721  024064                          TST354:
6722  024064  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6723  024066  012700  000354                  MOV     #354,R0         ;;LOAD R0 WITH TEST NUMBER
6724  024072  013701  024116                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6725  024076  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6726  024102  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
6727  024106  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6728  024110  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
6729  024114  000257                          CCC                     ;SCOPE SYNC
6730
6731  024116  105023                  2$:     CLRB    (R3)+           ;TEST THE CLRB
6732
6733  024120  022703  063237                  CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
6734  024124  001401                          BEQ     4$              ;BR IF YES
6735
6736  024126  104005                  3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
6737
6738  024130  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
6739  024132  001402                          BEQ     TST355          ;;BR IF YES
6740
6741  024134  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6742  024136  104001                  5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
6743
6744                                  ;;**********************************************************************
6745                                  ;*TEST 355       CLRB TEST - DM1 - EVEN ADDRESS
6746                                  ;;**********************************************************************
6747  024140                          TST355:
6748  024140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6749  024142  012700  000355                  MOV     #355,R0         ;;LOAD R0 WITH TEST NUMBER
6750  024146  013701  024172                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6751  024152  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6752  024156  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
6753  024162  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6754  024164  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
6755  024170  000257                          CCC                     ;SCOPE SYNC
6756
6757  024172  105013                  2$:     CLRB    (R3)            ;TEST THE CLRB
6758
6759  024174  020412                          CMP     R4,(R2)         ;CORRECT RESULT ?
6760  024176  001402                          BEQ     TST356          ;;BR IF YES
6761
6762  024200  011203                          MOV     (R2),R3         ;GET THE WAS DATA
6763  024202  104001                  3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
6764
6765                                  ;;**********************************************************************
6766                                  ;*TEST 356       NEGB TEST - DM2 - ODD ADDRESS
6767                                  ;;**********************************************************************
```

```
6768  024204                                  TST356:
6769  024204  000004                                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
6770  024206  012700  000356                          MOV     #356,R0                 ;;LOAD R0 WITH TEST NUMBER
6771  024212  013701  024242                          MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
6772  024216  012702  063237                          MOV     #MBUF0+1,R2             ;DEST ADDR = MBUF0+1
6773  024222  012704  000777                          MOV     #777,R4                 ;RESULT S / B = 777
6774  024226  012705  063236                          MOV     #MBUF0,R5               ;POINT R5 TO CHECK RESULT
6775  024232  010203                                  MOV     R2,R3                   ;R3 CONTAINS DEST ADDR
6776  024234  012715  177777                          MOV     #-1,(R5)                ;[DEST] = 177777
6777  024240  000257                                  CCC                             ;SCOPE SYNC
6778
6779  024242  105423                          2$:     NEGB    (R3)+                   ;TEST THE NEGB
6780
6781  024244  022703  063240                          CMP     #MBUF0+2,R3             ;DID DEST REG GET INCREMENTED ?
6782  024250  001401                                  BEQ     4$                      ;BR IF YES
6783
6784  024252  104005                          3$:     ERROR   5                       ;NEGB FAILED TO UPDATE DEST REG
6785
6786  024254  020415                          4$:     CMP     R4,(R5)                 ;CORRECT RESULT ?
6787  024256  001402                                  BEQ     TST357                  ;;BR IF YES
6788
6789  024260  011503                                  MOV     (R5),R3                 ;GET THE WAS DATA
6790  024262  104001                          5$:     ERROR   1                       ;NEGB DELIVERED WRONG RESULT
6791
6792                                          ;;**************************************************************
6793                                          ;*TEST 357      NEGB TEST - DM1 - ODD ADDRESS
6794                                          ;;**************************************************************
6795  024264                                  TST357:
6796  024264  000004                                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
6797  024266  012700  000357                          MOV     #357,R0                 ;;LOAD R0 WITH TEST NUMBER
6798  024272  013701  024334                          MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
6799                                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT7
6800  024276  032737  000200  063160                  BIT     #BIT7,@#BPTLOC          ;BREAKPOINT HALT SET ??
6801  024304  001401                                  BEQ     .+4                     ;BR IF NOT
6802  024306  000000                                  HALT                            ;BREAK - DEPRESS CONTINUE TO RESTART
6803  024310  012702  063237                          MOV     #MBUF0+1,R2             ;DEST ADDR = MBUF0+1
6804  024314  012704  000777                          MOV     #777,R4                 ;RESULT S / B = 777
6805  024320  012705  063236                          MOV     #MBUF0,R5               ;POINT R5 TO CHECK RESULT
6806  024324  010203                                  MOV     R2,R3                   ;R3 CONTAINS DEST ADDR
6807  024326  012715  177777                          MOV     #-1,(R5)                ;[DEST] = 177777
6808  024332  000257                                  CCC                             ;SCOPE SYNC
6809
6810  024334  105413                          2$:     NEGB    (R3)                    ;TEST THE NEGB
6811
6812  024336  020415                                  CMP     R4,(R5)                 ;CORRECT RESULT ?
6813  024340  001402                                  BEQ     TST360                  ;;BR IF YES
6814
6815  024342  011503                                  MOV     (R5),R3                 ;GET THE WAS DATA
6816  024344  104001                          3$:     ERROR   1                       ;NEGB DELIVERED WRONG RESULT
6817
6818                                          ;;**************************************************************
6819                                          ;*TEST 360      NEGB TEST - DM2 - EVEN ADDRESS
6820                                          ;;**************************************************************
6821  024346                                  TST360:
6822  024346  000004                                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
6823  024350  012700  000360                          MOV     #360,R0                 ;;LOAD R0 WITH TEST NUMBER
```

```
6824  024354  013701  024400          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6825  024360  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6826  024364  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
6827  024370  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6828  024372  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
6829  024376  000257                  CCC                     ;SCOPE SYNC
6830
6831  024400  105423          2$:     NEGB    (R3)+           ;TEST THE NEGB
6832
6833  024402  022703  063237          CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
6834  024406  001401                  BEQ     4$              ;BR IF YES
6835
6836  024410  104005          3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
6837
6838  024412  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
6839  024414  001402                  BEQ     TST361          ;;BR IF YES
6840
6841  024416  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6842  024420  104001          5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
6843
6844                          ;;**************************************************************
6845                          ;*TEST 361      NEGB TEST - DM1 - EVEN ADDRESS
6846                          ;;**************************************************************
6847  024422                  TST361:
6848  024422  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6849  024424  012700  000361          MOV     #361,R0         ;;LOAD R0 WITH TEST NUMBER
6850  024430  013701  024454          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6851  024434  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6852  024440  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
6853  024444  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
6854  024446  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
6855  024452  000257                  CCC                     ;SCOPE SYNC
6856
6857  024454  105413          2$:     NEGB    (R3)            ;TEST THE NEGB
6858
6859  024456  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
6860  024460  001402                  BEQ     TST362          ;;BR IF YES
6861
6862  024462  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6863  024464  104001          3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
6864
6865                          ;;**************************************************************
6866                          ;*TEST 362      ADD TEST - SM0,DM0 - N:C = 1010
6867                          ;;**************************************************************
6868  024466                  TST362:
6869  024466  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6870  024470  012700  000362          MOV     #362,R0         ;;LOAD R0 WITH TEST NUMBER
6871  024474  013701  024516          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6872  024500  005004                  CLR     R4              ;RESULT S / B = 000000
6873  024502  012705  177777          MOV     #-1,R5          ;SRC OPR = 177777
6874  024506  012703  000001          MOV     #+1,R3          ;[DEST] = +1
6875  024512  000257                  CCC                     ;CLEAR FLAGS
6876  024514  000272                  272                     ;N:C = 1010
6877
6878  024516  060503          2$:     ADD     R5,R3           ;TEST THE ADD
6879
```

```
6880  024520  100403                  BMI     3$              ;N:C = 0101
6881  024522  001002                  BNE     3$
6882  024524  102401                  BVS     3$
6883  024526  103401                  BCS     4$
6884
6885  024530  104002          3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
6886
6887  024532  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
6888  024534  001401                  BEQ     TST363          ;;BR IF YES
6889
6890  024536  104002          5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
6891
6892                          ;;**********************************************************************
6893                          ;*TEST 363      ADD TEST - SM0,DM0 - N:C = 0101
6894                          ;;**********************************************************************
6895  024540                  TST363:
6896  024540  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6897  024542  012700  000363          MOV     #363,R0         ;;LOAD R0 WITH TEST NUMBER
6898  024546  013701  024572          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6899  024552  012704  100006          MOV     #100006,R4      ;RESULT S / B = 100006
6900  024556  012705  077777          MOV     #77777,R5       ;SRC OPR = 77777
6901  024562  012703  000007          MOV     #7,R3           ;[DEST] = 7
6902  024566  000257                  CCC                     ;CLEAR FLAGS
6903  024570  000265                  265                     ;N:C = 0101
6904
6905  024572  060503          2$:     ADD     R5,R3           ;TEST THE ADD
6906
6907  024574  100003                  BPL     3$              ;N:C = 1010
6908  024576  001402                  BEQ     3$
6909  024600  102001                  BVC     3$
6910  024602  103001                  BCC     4$
6911
6912  024604  104002          3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
6913
6914  024606  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
6915  024610  001401                  BEQ     TST364          ;;BR IF YES
6916
6917  024612  104002          5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
6918
6919                          ;;**********************************************************************
6920                          ;*TEST 364      ADD SM1,DM0 TEST
6921                          ;;**********************************************************************
6922  024614                  TST364:
6923  024614  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6924  024616  012700  000364          MOV     #364,R0         ;;LOAD R0 WITH TEST NUMBER
6925  024622  013701  024642          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6926  024626  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
6927  024632  012705  063222          MOV     #ATA,R5         ;SOURCE ADDR = ATA
6928  024636  005003                  CLR     R3              ;[DEST] = 0
6929  024640  000257                  CCC                     ;SCOPE SYNC
6930
6931  024642  061503          2$:     ADD     (R5),R3         ;TEST THE ADD - SM1,DM0
6932
6933  024644  020403                  CMP     R4,R3           ;RESULT = #DWTA?
6934  024646  001401                  BEQ     4$              ;BR IF YES
6935
```

```
6936  024650  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
6937
6938  024652  022705  063222          4$:     CMP     #ATA,R5         ;DID ADD CHANGE REG.
6939  024656  001401                          BEQ     TST365          ;;BR IF NOT
6940
6941  024660  104005                  5$:     ERROR   5               ;REG GOT MODIFIED
6942
6943                                  ;;**************************************************************
6944                                  ;*TEST 365      ADD SM2,DMO TEST
6945                                  ;;**************************************************************
6946  024662                          TST365:
6947  024662  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6948  024664  012700  000365                  MOV     #365,R0         ;;LOAD R0 WITH TEST NUMBER
6949  024670  013701  024710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6950  024674  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
6951  024700  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
6952  024704  005003                          CLR     R3              ;[DEST] = 0
6953  024706  000257                          CCC                     ;SCOPE SYNC
6954
6955  024710  062503                  2$:     ADD     (R5)+,R3        ;TEST THE ADD - SM2,DMO
6956
6957  024712  020403                          CMP     R4,R3           ;RESULT = #DWTA
6958  024714  001401                          BEQ     4$              ;BR IF YES
6959
6960  024716  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
6961
6962  024720  022705  063224          4$:     CMP     #ATA+2,R5       ;DID ADD AUTO INCREMENT SOURCE REG?
6963  024724  001401                          BEQ     TST366          ;;BR IF YES
6964
6965  024726  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
6966
6967                                  ;;**************************************************************
6968                                  ;*TEST 366      ADD SM3,DMO TEST
6969                                  ;;**************************************************************
6970  024730                          TST366:
6971  024730  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6972  024732  012700  000366                  MOV     #366,R0         ;;LOAD R0 WITH TEST NUMBER
6973  024736  013701  024762                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6974  024742  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
6975  024746  012705  063232                  MOV     #ATA+10,R5      ;R5 POINTS TO SOURCE ADDR
6976  024752  010437  063236                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
6977  024756  005003                          CLR     R3              ;[DEST] = 0
6978  024760  000257                          CCC                     ;SCOPE SYNC
6979
6980  024762  063503                  2$:     ADD     @(R5)+,R3       ;TEST THE ADD - SM3,DMO
6981
6982  024764  020437  063236                  CMP     R4,@#MBUF0      ;RESULT = #DWTA?
6983  024770  001401                          BEQ     4$              ;BR IF YES
6984
6985  024772  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
6986
6987  024774  022705  063234          4$:     CMP     #ATA+12,R5      ;DID ADD AUTO INCREMENT SOURCE REG?
6988  025000  001401                          BEQ     TST367          ;;BR IF YES
6989
6990  025002  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
6991
```

```
                                ;;*****************************************************************
                                ;;TEST 367       ADD SM4,DMO TEST
                                ;;*****************************************************************
025004                          TST367:
025004  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025006  012700  000367                  MOV     #367,R0         ;;LOAD R0 WITH TEST NUMBER
025012  013701  025032                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025016  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025022  012705  063224                  MOV     #ATA+2,R5       ;SOURCE ADDR = ATA
025026  005003                          CLR     R3              ;[DEST] = 0
025030  000257                          CCC                     ;SCOPE SYNC

025032  064503                  2$:     ADD     -(R5),R3        ;TEST THE ADD - SM4,DMO

025034  020403                          CMP     R4,R3           ;RESULT = #DWTA?
025036  001401                          BEQ     4$              ;BR IF YES

025040  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

025042  022705  063222          4$:     CMP     #ATA,R5         ;DID SOURCE REG GET DECREMENTED?
025046  001401                          BEQ     TST370          ;;BR IF YES

025050  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG

                                ;;*****************************************************************
                                ;;TEST 370       ADD SM5,DMO TEST
                                ;;*****************************************************************
025052                          TST370:
025052  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025054  012700  000370                  MOV     #370,R0         ;;LOAD R0 WITH TEST NUMBER
025060  013701  025104                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025064  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025070  012705  063234                  MOV     #ATA+12,R5      ;R5 POINTS TO SOURCE ADDR
025074  010437  063236                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
025100  005003                          CLR     R3              ;[DEST] = 0
025102  000257                          CCC                     ;SCOPE SYNC

025104  065503                  2$:     ADD     @-(R5),R3       ;TEST THE ADD - SM5,DMO

025106  020437  063236                  CMP     R4,@#MBUF0      ;RESULT = #DWTA?
025112  001401                          BEQ     4$              ;BR IF YES

025114  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

025116  022705  063232          4$:     CMP     #ATA+10,R5      ;DID ADD DECREMENT SOURCE REG?
025122  001401                          BEQ     TST371          ;;BR IF YES

025124  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.

                                ;;*****************************************************************
                                ;;TEST 371       ADD SM6,DMO TEST
                                ;;*****************************************************************
025126                          TST371:
025126  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025130  012700  000371                  MOV     #371,R0         ;;LOAD R0 WITH TEST NUMBER
025134  013701  025154                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7048  025140  012704  063236              MOV     #MBUF0,R4       ;RESULT S / B = MBUF0
7049  025144  012705  063222              MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7050  025150  005003                      CLR     R3              ;[DEST] = 0
7051  025152  000257                      CCC                     ;SCOPE SYNC
7052
7053  025154  066503  000010      2$:     ADD     10(R5),R3       ;TEST THE ADD - SM6,DM0
7054
7055  025160  020403                      CMP     R4,R3           ;RESULT =MBUF0?
7056  025162  001401                      BEQ     TST372          ;;BR IF YES
7057
7058  025164  104002              3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7059
7060                              ;;**********************************************************************
7061                              ;*TEST 372       ADD SM7,DM0 TEST
7062                              ;;**********************************************************************
7063  025166                      TST372:
7064  025166  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7065  025170  012700  000372              MOV     #372,R0         ;;LOAD R0 WITH TEST NUMBER
7066  025174  013701  025220              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7067  025200  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7068  025204  012705  063222              MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
7069  025210  010437  063236              MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7070  025214  005003                      CLR     R3              ;[DEST] = 0
7071  025216  000257                      CCC                     ;SCOPE SYNC
7072
7073  025220  067503  000C1C      2$:     ADD     @10(R5),R3      ;TEST THE ADD - SM7,DM0
7074
7075  025224  020403                      CMP     R4,R3           ;RESULT = #DWTA?
7076  025226  001401                      BEQ     TST373          ;;BR IF YES
7077
7078  025230  104002              3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7079
7080                              ;;**********************************************************************
7081                              ;*TEST 373       ADD SM1,DM1 TEST
7082                              ;;**********************************************************************
7083  025232                      TST373:
7084  025232  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7085  025234  012700  000373              MOV     #373,R0         ;;LOAD R0 WITH TEST NUMBER
7086  025240  013701  025264              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7087  025244  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7088  025250  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7089  025254  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7090  025260  005012                      CLR     (R2)            ;[DEST] = 0
7091  025262  000257                      CCC                     ;SCOPE SYNC
7092
7093  025264  061512              2$:     ADD     (R5),(R2)       ;TEST THE ADD - SM1,DM1
7094
7095  025266  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7096  025270  001402                      BEQ     TST374          ;;BR IF YES
7097
7098  025272  011203                      MOV     (R2),R3         ;GET WAS DATA
7099  025274  104001              3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7100
7101                              ;;**********************************************************************
7102                              ;*TEST 374       ADD SM2,DM1 TEST
7103                              ;;**********************************************************************
```

```
7104  025276                          TST374:
7105  025276  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7106  025300  012700  000374                  MOV     #374,R0         ;;LOAD R0 WITH TEST NUMBER
7107  025304  013701  025330                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7108  025310  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7109  025314  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7110  025320  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7111  025324  005012                          CLR     (R2)            ;[DEST] = 0
7112  025326  000257                          CCC                     ;SCOPE SYNC
7113
7114  025330  062512                  2$:     ADD     (R5)+,(R2)      ;TEST THE ADD - SM2,DM1
7115
7116  025332  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
7117  025334  001402                          BEQ     TST375          ;;BR IF YES
7118
7119  025336  011203                          MOV     (R2),R3         ;GET WAS DATA
7120  025340  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7121
7122                                  ;;**********************************************************************
7123                                  ;*TEST 375       ADD SM1,DM2 TEST
7124                                  ;;**********************************************************************
7125  025342                          TST375:
7126  025342  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7127  025344  012700  000375                  MOV     #375,R0         ;;LOAD R0 WITH TEST NUMBER
7128  025350  013701  025376                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7129  025354  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7130  025360  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7131  025364  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
7132  025370  010203                          MOV     R2,R3           ;[R3] = DEST ADDR
7133  025372  005012                          CLR     (R2)            ;[DEST] = 0
7134  025374  000257                          CCC                     ;SCOPE SYNC
7135
7136  025376  061523                  2$:     ADD     (R5),(R3)+      ;TEST THE ADD - SM1,DM2
7137
7138  025400  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
7139  025402  001406                          BEQ     4$              ;BR IF YES
7140
7141  025404  010337  063242                  MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
7142  025410  011203                          MOV     (R2),R3         ;GET WAS DATA
7143  025412  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7144
7145  025414  013703  063242                  MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
7146  025420  022703  063240          4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG
7147  025424  001401                          BEQ     TST376          ;;BR IF YES
7148
7149  025426  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7150
7151                                  ;;**********************************************************************
7152                                  ;*TEST 376       ADD SM2,DM2 TEST
7153                                  ;;**********************************************************************
7154  025430                          TST376:
7155  025430  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7156  025432  012700  000376                  MOV     #376,R0         ;;LOAD R0 WITH TEST NUMBER
7157  025436  013701  025464                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7158  025442  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7159  025446  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
```

```
 7160  025452  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
 7161  025456  010203                          MOV     R2,R3           ;[R3] = DEST ADDR
 7162  025460  005012                          CLR     (R2)            ;[DEST] = 0
 7163  025462  000257                          CCC                     ;SCOPE SYNC
 7164
 7165  025464  062523              2$:         ADD     (R5)+,(R3)+     ;TEST THE ADD - SM2,DM2
 7166
 7167  025466  020412                          CMP     R4,(R2)         ;RESULT = #DWTA
 7168  025470  001406                          BEQ     4$              ;BR IF YES
 7169
 7170  025472  010337  063242                  MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
 7171  025476  011203                          MOV     (R2),R3         ;GET WAS DATA
 7172  025500  104001              3$:         ERROR   1               ;ADD DELIVERED WRONG RESULT
 7173
 7174  025502  013703  063242                  MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
 7175  025506  022703  063240      4$:         CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG?
 7176  025512  001401                          BEQ     TST377          ;;BR IF YES
 7177
 7178  025514  104005              5$:         ERROR   5               ;ADD FAILED TO UPDATE DEST REG
 7179
 7180                              ;;**************************************************************
 7181                              ;*TEST 377       ADD SM1,DM3 TEST
 7182                              ;;**************************************************************
 7183  025516                      TST377:
 7184  025516  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7185  025520  012700  000377                  MOV     #377,R0         ;;LOAD R0 WITH TEST NUMBER
 7186  025524  013701  025554                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7187  025530  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7188  025534  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 7189  025540  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
 7190  025544  012703  063232                  MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
 7191  025550  005012                          CLR     (R2)            ;[DEST] = 0
 7192  025552  000257                          CCC                     ;SCOPE SYNC
 7193
 7194  025554  061533              2$:         ADD     (R5),@(R3)+     ;TEST THE ADD - SM1,DM3
 7195
 7196  025556  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
 7197  025560  001406                          BEQ     4$              ;BR IF YES
 7198
 7199  025562  010337  063242                  MOV     R3,@#MBUF1      ;SAVE R3
 7200  025566  011203                          MOV     (R2),R3         ;GET WAS DATA
 7201  025570  104001              3$:         ERROR   1               ;ADD DELIVERED WRONG RESULT
 7202
 7203  025572  013703  063242                  MOV     @#MBUF1,R3      ;RESTORE R3
 7204  025576  022703  063234      4$:         CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
 7205  025602  001401                          BEQ     TST400          ;;BR IF YES
 7206
 7207  025604  104005              5$:         ERROR   5               ;ADD FAILED TO UPDATE DEST REG
 7208
 7209                              ;;**************************************************************
 7210                              ;*TEST 400       ADD SM2,DM3 TEST
 7211                              ;;**************************************************************
 7212  025606                      TST400:
 7213  025606  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7214  025610  012700  000400                  MOV     #400,R0         ;;LOAD R0 WITH TEST NUMBER
 7215  025614  013701  025644                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7216  025620  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7217  025624  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7218  025630  012705  063222          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7219  025634  012703  063232          MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
7220  025640  005012                  CLR     (R2)            ;[DEST] = 0
7221  025642  000257                  CCC                     ;SCOPE SYNC
7222
7223  025644  062533          2$:     ADD     (R5)+,@(R3)+    ;TEST THE ADD - SM2,DM3
7224
7225  025646  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7226  025650  001406                  BEQ     4$              ;BR IF YES
7227
7228  025652  010337  063242          MOV     R3,@#MBUF1      ;SAVE R3
7229  025656  011203                  MOV     (R2),R3         ;GET WAS DATA
7230  025660  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7231
7232  025662  013703  063242          MOV     @#MBUF1,R3      ;RESTORE R3
7233  025666  022703  063234  4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
7234  025672  001401                  BEQ     TST4C1          ;;BR IF YES
7235
7236  025674  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7237
7238                          ;;**********************************************************************
7239                          ;*TEST 401      ADD SM1,DM4 TEST
7240                          ;;**********************************************************************
7241  025676                  TST401:
7242  025676  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7243  025700  012700  000401          MOV     #401,R0         ;;LOAD R0 WITH TEST NUMBER
7244  025704  013701  025734          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7245  025710  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7246  025714  012704  063246          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7247  025720  012705  063222          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7248  025724  012703  063240          MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
7249  025730  005012                  CLR     (R2)            ;[DEST] = 0
7250  025732  000257                  CCC                     ;SCOPE SYNC
7251
7252  025734  061543          2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM1,DM4
7253
7254  025736  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
7255  025740  001406                  BEQ     4$              ;BR IF YES
7256
7257  025742  010337  063242          MOV     R3,@#MBUF1      ;SAVE R3
7258  025746  011203                  MOV     (R2),R3         ;GET WAS DATA
7259  025750  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7260
7261  025752  013703  063242          MOV     @#MBUF1,R3      ;RESTORE R3
7262  025756  020302          4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
7263  025760  001401                  BEQ     TST402          ;;BR IF YES
7264
7265  025762  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
7266
7267                          ;;**********************************************************************
7268                          ;*TEST 402      ADD SM2,DM4 TEST
7269                          ;;**********************************************************************
7270  025764                  TST402:
7271  025764  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
7272  025766  012700  000402              MOV     #402,R0         ;;LOAD R0 WITH TEST NUMBER
7273  025772  013701  026022              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7274  025776  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7275  026002  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7276  026006  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7277  026012  012703  063240              MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
7278  026016  005012                      CLR     (R2)            ;[DEST] = 0
7279  026020  000257                      CCC                     ;SCOPE SYNC
7280
7281  026022  061543              2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM2,DM4
7282
7283  026024  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7284  026026  001406                      BEQ     4$              ;BR IF YES
7285
7286  026030  010337  063242              MOV     R3,@#MBUF1      ;SAVE R3
7287  026034  011203                      MOV     (R2),R3         ;GET WAS DATA
7288  026036  104001              3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7289
7290  026040  013703  063242              MOV     @#MBUF1,R3      ;RESTORE R3
7291  026044  020302              4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
7292  026046  001401                      BEQ     TST403          ;;BR IF YES
7293
7294  026050  104005              5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
7295
7296                              ;;**************************************************************
7297                              ;*TEST 403      ADD SM1,DM5 TEST
7298                              ;;**************************************************************
7299  026052                      TST403:
7300  026052  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7301  026054  012700  000403              MOV     #403,R0         ;;LOAD R0 WITH TEST NUMBER
7302  026060  013701  026110              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7303  026064  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7304  026070  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7305  026074  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7306  026100  012703  063234              MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
7307  026104  005012                      CLR     (R2)            ;[DEST] = 0
7308  026106  000257                      CCC                     ;SCOPE SYNC
7309
7310  026110  061553              2$:     ADD     (R5),@-(R3)     ;TEST THE ADD - SM1,DM5
7311
7312  026112  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7313  026114  001406                      BEQ     4$              ;BR IF YES
7314
7315  026116  010337  063242              MOV     R3,@#MBUF1      ;SAVE R3
7316  026122  011203                      MOV     (R2),R3         ;GET WAS DATA
7317  026124  104001              3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7318
7319  026126  013703  063242              MOV     @#MBUF1,R3      ;RESTORE R3
7320  026132  022703  063232              CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
7321  026136  001401                      BEQ     TST404          ;;BR IF YES
7322
7323  026140  104005              5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
7324
7325                              ;;**************************************************************
7326                              ;*TEST 404      ADD SM2,DM5 TEST
7327                              ;;**************************************************************
```

```
 7328  026142                          TST404:
 7329  026142  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7330  026144  012700  000404                  MOV     #404,R0         ;;LOAD R0 WITH TEST NUMBER
 7331  026150  013701  026200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7332  026154  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7333  026160  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 7334  026164  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
 7335  026170  012703  063234                  MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
 7336  026174  005012                          CLR     (R2)            ;[DEST] = 0
 7337  026176  000257                          CCC                     ;SCOPE SYNC
 7338
 7339  026200  062553                  2$:     ADD     (R5)+,@-(R3)    ;TEST THE ADD - SM2,DM5
 7340
 7341  026202  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
 7342  026204  001406                          BEQ     4$              ;BR IF YES
 7343
 7344  026206  010337  063242                  MOV     R3,@#MBUF1      ;SAVE R3
 7345  026212  011203                          MOV     (R2),R3         ;GET WAS DATA
 7346  026214  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
 7347
 7348  026216  013703  063242                  MOV     @#MBUF1,R3      ;RESTORE R3
 7349  026222  022703  063232          4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
 7350  026226  001401                          BEQ     TST405          ;;BR IF YES
 7351
 7352  026230  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
 7353
 7354                                  ;;**************************************************************
 7355                                  ;*TEST 405      ADD SM1,DM6 TEST
 7356                                  ;;**************************************************************
 7357  026232                          TST405:
 7358  026232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7359  026234  012700  000405                  MOV     #405,R0         ;;LOAD R0 WITH TEST NUMBER
 7360  026240  013701  026270                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7361  026244  012702  063242                  MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
 7362  026250  012704  063246                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 7363  026254  012705  063222                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
 7364  026260  012703  063236                  MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
 7365  026264  005012                          CLR     (R2)            ;[DEST] = 0
 7366  026266  000257                          CCC                     ;SCOPE SYNC
 7367
 7368  026270  061563  000004          2$:     ADD     (R5),4(R3)      ;TEST THE ADD - SM1,DM6
 7369
 7370  026274  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
 7371  026276  001402                          BEQ     TST406          ;;BR IF YES
 7372
 7373  026300  011203                          MOV     (R2),R3         ;GET WAS DATA
 7374  026302  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
 7375
 7376                                  ;;**************************************************************
 7377                                  ;*TEST 406      ADD SM2,DM6 TEST
 7378                                  ;;**************************************************************
 7379  026304                          TST406:
 7380  026304  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7381  026306  012700  000406                  MOV     #406,R0         ;;LOAD R0 WITH TEST NUMBER
 7382  026312  013701  026342                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7383  026316  012702  063242                  MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
```

```
7384  026322  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7385  026326  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7386  026332  012703  063236              MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
7387  026336  005012                      CLR     (R2)            ;[DEST] = 0
7388  026340  000257                      CCC                     ;SCOPE SYNC
7389
7390  026342  062563  000004      2$:     ADD     (R5)+,4(R3)     ;TEST THE ADD - SM2,DM6
7391
7392  026346  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7393  026350  001402                      BEQ     TST407          ;;BR IF YES
7394
7395  026352  011203                      MOV     (R2),R3         ;GET WAS DATA
7396  026354  104001              3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7397
7398                              ;;**************************************************************
7399                              ;*TEST 407       ADD SM1,DM7 TEST
7400                              ;;**************************************************************
7401  026356                      TST407:
7402  026356  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7403  026360  012700  000407              MOV     #407,R0         ;;LOAD R0 WITH TEST NUMBER
7404  026364  013701  026412              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7405  026370  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7406  026374  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7407  026400  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7408  026404  010503                      MOV     R5,R3           ;BASE DEST ADDR = ATA
7409  026406  005012                      CLR     (R2)            ;[DEST] = 0
7410  026410  000257                      CCC                     ;SCOPE SYNC
7411
7412  026412  061573  000010      2$:     ADD     (R5),@10(R3)    ;TEST THE ADD - SM1,DM7
7413
7414  026416  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7415  026420  001402                      BEQ     TST410          ;;BR IF YES
7416
7417  026422  011203                      MOV     (R2),R3         ;GET WAS DATA
7418  026424  104001              3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7419
7420                              ;;**************************************************************
7421                              ;*TEST 410       ADD SM2,DM7 TEST
7422                              ;;**************************************************************
7423  026426                      TST410:
7424  026426  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7425  026430  012700  000410              MOV     #410,R0         ;;LOAD R0 WITH TEST NUMBER
7426  026434  013701  026462              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7427  026440  012702  063236              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7428  026444  012704  063246              MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7429  026450  012705  063222              MOV     #ATA,R5         ;SOURCE ADDR = ATA
7430  026454  010503                      MOV     R5,R3           ;BASE DEST ADDR = ATA
7431  026456  005012                      CLR     (R2)            ;[DEST] = 0
7432  026460  000257                      CCC                     ;SCOPE SYNC
7433
7434  026462  062573  000010      2$:     ADD     (R5)+,@10(R3)   ;TEST THE ADD - SM2,DM7
7435
7436  026466  020412                      CMP     R4,(R2)         ;RESULT = #DWTA?
7437  026470  001402                      BEQ     TST411          ;;BR IF YES
7438
7439  026472  011203                      MOV     (R2),R3         ;GET WAS DATA
```

```
7440  026474  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
7441
7442                                  ;;*****************************************************************
7443                                  ;*TEST 411      "XOR RA,RB" TEST - A=B=000000 N:C=1010
7444                                  ;;*****************************************************************
7445  026476                          TST411:
7446  026476  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7447  026500  012700  000411                 MOV     #411,R0         ;;LOAD R0 WITH TEST NUMBER
7448  026504  013701  026520                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7449  026510  005004                         CLR     R4              ;RESULT AND MASK = 000000
7450  026512  005003                         CLR     R3              ;[DEST] = 000000
7451  026514  000257                         CCC                     ;SCOPE SYNC
7452  026516  000272                         272                     ;MAKE N:C=1010
7453
7454  026520  074403                  2$:     XOR     R4,R3           ;TEST THE XOR
7455
7456  026522  100403                         BMI     3$              ;N:C=0100 ??
7457  026524  001002                         BNE     3$
7458  026526  102401                         BVS     3$
7459  026530  103001                         BCC     4$
7460
7461  026532  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
7462
7463  026534  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
7464  026536  001401                         BEQ     TST412          ;;BR IF YES
7465
7466  026540  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
7467
7468                                  ;;*****************************************************************
7469                                  ;*TEST 412      "XOR RA,RB" TEST - A=B=177777 N:C=0101
7470                                  ;;*****************************************************************
7471  026542                          TST412:
7472  026542  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7473  026544  012700  000412                 MOV     #412,R0         ;;LOAD R0 WITH TEST NUMBER
7474  026550  013701  026570                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7475  026554  005004                         CLR     R4              ;RESULT  = 000000
7476  026556  012705  177777                 MOV     #-1,R5          ;MASK = 177777
7477  026562  010503                         MOV     R5,R3           ;[DEST]=177777
7478  026564  000257                         CCC                     ;SCOPE SYNC
7479  026566  000265                         265                     ;MAKE N:C=0101
7480
7481  026570  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
7482
7483  026572  100403                         BMI     3$              ;N:C=0101 ??
7484  026574  001002                         BNE     3$
7485  026576  102401                         BVS     3$
7486  026600  103401                         BCS     4$
7487
7488  026602  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
7489
7490  026604  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
7491  026606  001401                         BEQ     TST413          ;;BR IF YES
7492
7493  026610  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
7494
7495                                  ;;*****************************************************************
```

```
MAINDEC-11-DQKDA-A   KD11-K BASIC LOGIC TESTS   MACY11 27(1006)  08-FEB-77  16:23  PAGE 138
DQKDAA.P11    08-FEB-77 16:17                   T413    "XOR RA,RB" TEST - A=125252,B=052525 N:C=0110

                                                ;*TEST 413      "XOR RA,RB" TEST - A=125252,B=052525 N:C=0110
                                                ;;*****************************************************************
026612                                          TST413:
026612  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026614  012700  000413                                 MOV     #413,R0         ;;LOAD R0 WITH TEST NUMBER
026620  013701  026644                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026624  012704  177777                                 MOV     #-1,R4          ;RESULT S/B = 177777
026630  012705  125252                                 MOV     #125252,R5      ;MASK=125252
026634  012703  052525                                 MOV     #052525,R3      ;[DEST] = 052525
026640  000257                                         CCC                     ;SCOPE SYNC
026642  000266                                         266                     ;MAKE N:C=0'10

026644  074503                                 2$:     XOR     R5,R3           ;TEST THE XOR

026646  100003                                         BPL     3$              ;N:C=1000 ??
026650  001402                                         BEQ     3$
026652  102401                                         BVS     3$
026654  103001                                         BCC     4$

026656  104002                                 3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY

026660  020403                                 4$:     CMP     R4,R3           ;RESULT CORRECT?
026662  001401                                         BEQ     TST414          ;;BR IF YES

026664  104002                                 5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT

                                                ;;*****************************************************************
                                                ;*TEST 414      "XOR RA,RB" TEST - A=052525,B=125252 N:C=1001
                                                ;;*****************************************************************
026666                                          TST414:
026666  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026670  012700  000414                                 MOV     #414,R0         ;;LOAD R0 WITH TEST NUMBER
026674  013701  026720                                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026700  012704  177777                                 MOV     #-1,R4          ;RESULT S/B = 177777
026704  012705  052525                                 MOV     #52525,R5       ;MASK=052525
026710  012703  125252                                 MOV     #125252,R3      ;[DEST] = 125252
026714  000257                                         CCC                     ;SCOPE SYNC
026716  000271                                         271                     ;MAKE N:C=1001

026720  074503                                 2$:     XOR     R5,R3           ;TEST THE XOR

026722  100003                                         BPL     3$              ;N:C=1001 ??
026724  001402                                         BEQ     3$
026726  102401                                         BVS     3$
026730  103401                                         BCS     4$

026732  104002                                 3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY

026734  020403                                 4$:     CMP     R4,R3           ;RESULT CORRECT?
026736  001401                                         BEQ     TST415          ;;BR IF YES

026740  104002                                 5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT

                                                ;;*****************************************************************
                                                ;*TEST 415      "XOR RA,(RB)" TEST - A=B=000000 N:C=1010
                                                ;;*****************************************************************
```

```
7552  026742                          TST415:
7553  026742  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7554  026744  012700  000415                  MOV     #415,R0         ;;LOAD R0 WITH TEST NUMBER
7555  026750  013701  026772                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7556  026754  005004                          CLR     R4              ;RESULT S / B = 000000
7557  026756  005005                          CLR     R5              ;MASK = 000000
7558  026760  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7559  026764  005012                          CLR     (R2)            ;[DEST] = 000000
7560  026766  000257                          CCC                     ;SCOPE SYNC
7561  026770  000272                          272                     ;MAKE N:C=1010
7562
7563  026772  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
7564
7565  026774  100403                          BMI     3$              ;N:C = 0100 ??
7566  026776  001002                          BNE     3$
7567  027000  102401                          BVS     3$
7568  027002  103001                          BCC     4$
7569
7570  027004  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
7571
7572  027006  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
7573  027010  001402                          BEQ     TST416          ;;BR IF YES
7574
7575  027012  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7576  027014  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
7577
7578                                  ;;**************************************************************
7579                                  ;TEST 416       "XOR RA,(RB)" TEST - A=B=177777 N:C=0101
7580                                  ;;**************************************************************
7581  027016                          TST416:
7582  027016  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7583  027020  012700  000416                  MOV     #416,R0         ;;LOAD R0 WITH TEST NUMBER
7584  027024  013701  027052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7585  027030  005004                          CLR     R4              ;RESULT S / B = 000000
7586  027032  012705  177777                  MOV     #-1,R5          ;MASK = 177777
7587  027036  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7588  027042  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
7589  027046  000257                          CCC                     ;SCOPE SYNC
7590  027050  000265                          265                     ;MAKE N:C=0101
7591
7592  027052  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
7593
7594  027054  100403                          BMI     3$              ;N:C = 0101 ??
7595  027056  001002                          BNE     3$
7596  027060  102401                          BVS     3$
7597  027062  103401                          BCS     4$
7598
7599  027064  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
7600
7601  027066  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
7602  027070  001402                          BEQ     TST417          ;;BR IF YES
7603
7604  027072  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7605  027074  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
7606
7607                                  ;;**************************************************************
```

```
                                        ;*TEST 417      "XOR RA,(RB)" TEST - A=125252,B=052525 N:C=0110
                                        ;;**************************************************************
027076                                  TST417:
027076  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027100  012700  000417                          MOV     #417,R0         ;;LOAD R0 WITH TEST NUMBER
027104  013701  027146                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        .SBTTL USER CONTROLLED BREAKPOINT -- BIT8
027110  032737  000400  063160                  BIT     #BIT8,@#BPTLOC  ;BREAKPOINT HALT SET ??
027116  001401                                  BEQ     .+4             ;BR IF NOT
027120  000000                                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
027122  012704  177777                          MOV     #-1,R4          ;RESULT S/B = 177777
027126  012705  125252                          MOV     #125252,R5      ;MASK = 125252
027132  012702  063236                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027136  012712  052525                          MOV     #052525,(R2)    ;[DEST] = 052525
027142  000257                                  CCC                     ;SCOPE SYNC
027144  000266                                  266                     ;MAKE N:C=0110
                                        
027146  074512                        2$:       XOR     R5,(R2)         ;TEST THE XOR
                                        
027150  100003                                  BPL     3$              ;N:C = 1000 ??
027152  001402                                  BEQ     3$
027154  102401                                  BVS     3$
027156  103001                                  BCC     4$
                                        
027160  104001                        3$:       ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
                                        
027162  020412                        4$:       CMP     R4,(R2)         ;RESULT CORRECT?
027164  001402                                  BEQ     TST420          ;;BR IF YES
                                        
027166  011203                                  MOV     (R2),R3         ;GET THE WAS DATA
027170  104001                        5$:       ERROR   1               ;XOR DELIVERED THE WRONG RESULT
                                        
                                        ;;**************************************************************
                                        ;*TEST 420      "XOR RA,(RB)" TEST - A=052525,B=125252 N:C=1001
                                        ;;**************************************************************
027172                                  TST420:
027172  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027174  012700  000420                          MOV     #420,R0         ;;LOAD R0 WITH TEST NUMBER
027200  013701  027230                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027204  012704  177777                          MOV     #-1,R4          ;RESULT S/B = 177777
027210  012705  052525                          MOV     #52525,R5       ;MASK = 052525
027214  012702  063236                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027220  012712  125252                          MOV     #125252,(R2)    ;[DEST] = 125252
027224  000257                                  CCC                     ;SCOPE SYNC
027226  000271                                  271                     ;MAKE N:C=1001
                                        
027230  074512                        2$:       XOR     R5,(R2)         ;TEST THE XOR
                                        
027232  100003                                  BPL     3$              ;N:C = 1001 ??
027234  001402                                  BEQ     3$
027236  102401                                  BVS     3$
027240  103401                                  BCS     4$
                                        
027242  104001                        3$:       ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
                                        
027244  020412                        4$:       CMP     R4,(R2)         ;RESULT CORRECT?
```

```
7664  027246  001402                    BEQ     TST421          ;;BR IF YES
7665
7666  027250  011203                    MOV     (R2),R3         ;GET THE WAS DATA
7667  027252  104001            5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
7668
7669                              ;;**********************************************************************
7670                              ;*TEST 421       SUB TEST SMD,DMD - (SRC) = (DEST) = +,+
7671                              ;;**********************************************************************
7672  027254                      TST421:
7673  027254  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7674  027256  012700  000421            MOV     #421,R0         ;;LOAD R0 WITH TEST NUMBER
7675  027262  013701  027302            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7676  027266  005004                    CLR     R4              ;RESULT S / B = 0
7677  027270  012703  052525            MOV     #052525,R3      ;[R3] = DEST OP = 52525
7678  027274  010305                    MOV     R3,R5           ;[R5] = SRC OP = 52525
7679  027276  000257                    CCC                     ;CLEAR FLAGS
7680  027300  000273                    273                     ;MAKE N:C = 1011
7681
7682  027302  160503            2$:     SUB     R5,R3           ;TEST THE SUB
7683
7684  027304  100403                    BMI     3$
7685  027306  001002                    BNE     3$              ;DID N:C = 0100
7686  027310  102401                    BVS     3$
7687  027312  103001                    BCC     4$
7688
7689  027314  104002            3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7690
7691  027316  020304            4$:     CMP     R3,R4           ;WAS RESULT = 0?
7692  027320  001401                    BEQ     TST422          ;;BR IF YES
7693
7694  027322  104002            5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
7695
7696
7697                              ;;**********************************************************************
7698                              ;*TEST 422       SUB TEST SMD,DMD - (SRC) = (DEST) = -,-
7699                              ;;**********************************************************************
7700  027324                      TST422:
7701  027324  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7702  027326  012700  000422            MOV     #422,R0         ;;LOAD R0 WITH TEST NUMBER
7703  027332  013701  027352            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7704  027336  005004                    CLR     R4              ;RESULT S / B = 0
7705  027340  012703  125252            MOV     #125252,R3      ;[R3] = DEST OP = 125252
7706  027344  010305                    MOV     R3,R5           ;[R5] = SOURCE OP = 125252
7707  027346  000257                    CCC                     ;CLEAR FLAGS
7708  027350  000273                    273                     ;MAKE N:C = 1011
7709  027352  160503            2$:     SUB     R5,R3           ;TEST THE SUB
7710
7711  027354  100403                    BMI     3$
7712  027356  001002                    BNE     3$              ;N:C = 0100?
7713  027360  102401                    BVS     3$
7714  027362  103001                    BCC     4$
7715
7716  027364  104002            3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7717
7718  027366  020304            4$:     CMP     R3,R4           ;RESULT = 0?
7719  027370  001401                    BEQ     TST423          ;;BR IF YES
```

```
7720
7721  027372  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
7722
7723                                  ;;*****************************************************************
7724                                  ;*TEST 423      SUB TEST SMO,DMO - (SRC) = (DEST) = -,+
7725                                  ;;*****************************************************************
7726  027374                          TST423:
7727  027374  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7728  027376  012700  000423                  MOV     #423,R0         ;;LOAD R0 WITH TEST NUMBER
7729  027402  013701  027426                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7730  027406  012704  000002                  MOV     #2,R4           ;RESULT S / B = 2
7731  027412  012703  000001                  MOV     #1,R3           ;[R3] = DEST OP = 1
7732  027416  012705  177777                  MOV     #-1,R5          ;[R5] = SRC OP = -1
7733  027422  000257                          CCC                     ;CLEAR FLAGS
7734  027424  000276                          276                     ;MAKE N:C = 1110
7735
7736  027426  160503                  2$:     SUB     R5,R3           ;TEST THE SUB
7737
7738  027430  100403                          BMI     3$
7739  027432  001402                          BEQ     3$              ;N:C = 0001
7740  027434  102401                          BVS     3$
7741  027436  103401                          BCS     4$
7742
7743  027440  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7744
7745  027442  020304                  4$:     CMP     R3,R4           ;RESULT = +2?
7746  027444  001401                          BEQ     TST424          ;;BR IF YES
7747
7748  027446  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
7749
7750                                  ;;*****************************************************************
7751                                  ;*TEST 424      SUB TEST SMO,DMO (SRC) = -(DEST) = +,-
7752                                  ;;*****************************************************************
7753  027450                          TST424:
7754  027450  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7755  027452  012700  000424                  MOV     #424,R0         ;;LOAD R0 WITH TEST NUMBER
7756  027456  013701  027502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7757  027462  012704  177776                  MOV     #-2,R4          ;RESULT S / B = -2
7758  027466  012703  177777                  MOV     #-1,R3          ;[R3] = [DEST] = -1
7759  027472  012705  000001                  MOV     #1,R5           ;[R5] = [SOURCE] = +1
7760  027476  000257                          CCC                     ;CLEAR FLAGS
7761  027500  000267                          267                     ;MAKE N:C = 0111
7762
7763  027502  160503                  2$:     SUB     R5,R3           ;TEST THE SUB
7764
7765  027504  100003                          BPL     3$
7766  027506  001402                          BEQ     3$              ;N:C = 1000
7767  027510  102401                          BVS     3$
7768  027512  103001                          BCC     4$
7769
7770  027514  104002                  3$:     ERROR   2               ;SUB DID NOT ALTER CODES PROPERLY
7771
7772  027516  020403                  4$:     CMP     R4,R3           ;RESULT = -2?
7773  027520  001401                          BEQ     TST425          ;;BR IF YES
7774
7775  027522  104002                  5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
```

```
7776
7777                                ;;*********************************************************************
7778                                ;*TEST 425      SUB TEST SM0,DM0 - "V" BIT SETS
7779                                ;;*********************************************************************
7780   027524                       TST425:
7781   027524  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7782   027526  012700  000425               MOV     #425,R0         ;;LOAD R0 WITH TEST NUMBER
7783   027532  013701  027556               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7784   027536  012704  077777               MOV     #77777,R4       ;RESULT = 77777
7785   027542  012703  100000               MOV     #100000,R3      ;[R3] = DEST OP = 100000
7786   027546  012705  000001               MOV     #1,R5           ;[R5] = SRC OP = 1
7787   027552  000257                       CCC                     ;CLEAR FLAGS
7788   027554  000274                       274                     ;MAKE N:C = 1100
7789
7790   027556  160503               2$:     SUB     R5,R3           ;TEST THE SUB
7791
7792   027560  100403                       BMI     3$
7793   027562  001402                       BEQ     3$              ;N:C = 0011 ("V" BIT SHOULD SET)
7794   027564  102001                       BVC     3$
7795   027566  103001                       BCC     4$
7796
7797   027570  104002               3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7798
7799   027572  020304               4$:     CMP     R3,R4           ;RESULT = 77777?
7800   027574  001401                       BEQ     TST426          ;;BR IF YES
7801
7802   027576  104002               5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
7803
7804                                ;;*********************************************************************
7805                                ;*TEST 426      SUB TEST - SM0,DM1 - N:C = 0110
7806                                ;;*********************************************************************
7807   027600                       TST426:
7808   027600  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7809   027602  012700  000426               MOV     #426,R0         ;;LOAD R0 WITH TEST NUMBER
7810   027606  013701  027634               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7811   027612  012702  063236               MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7812   027616  012704  177777               MOV     #-1,R4          ;RESULT S / B = 177777
7813   027622  012705  000001               MOV     #+1,R5          ;SRC OPR = +1
7814   027626  005012                       CLR     (R2)            ;[DEST] = 000000
7815   027630  000257                       CCC                     ;CLEAR FLAGS
7816   027632  000266                       266                     ;N:C = 0110
7817
7818   027634  160512               2$:     SUB     R5,(R2)         ;TEST THE SUB
7819
7820   027636  100003                       BPL     3$              ;N:C = 1001
7821   027640  001402                       BEQ     3$
7822   027642  102401                       BVS     3$
7823   027644  103401                       BCS     4$
7824
7825   027646  104001               3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
7826
7827   027650  020412               4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7828   027652  001402                       BEQ     TST427          ;;BR IF YES
7829
7830   027654  011203                       MOV     (R2),R3         ;GET THE WAS DATA
7831   027656  104001               5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 427      SUB TEST - SM0,DM1 - N:C = 1010
                                ;;*****************************************************************
027660                          TST427:
027660  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027662  012700  000427                MOV     #427,R0         ;;LOAD R0 WITH TEST NUMBER
027666  013701  027714                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027672  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027676  005004                        CLR     R4              ;RESULT S / B = 000000
027700  012705  177777                MOV     #-1,R5          ;SRC OPR = 177777
027704  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
027710  000257                        CCC                     ;CLEAR FLAGS
027712  000272                        272                     ;N:C = 1010
                                
027714  160512                  2$:   SUB     R5,(R2)         ;TEST THE SUB
                                
027716  100403                        BMI     3$              ;N:C = 0100
027720  001002                        BNE     3$
027722  102401                        BVS     3$
027724  103001                        BCC     4$
                                
027726  104001                  3$:   ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
                                
027730  020412                  4$:   CMP     R4,(R2)         ;CORRECT RESULT ?
027732  001402                        BEQ     TST430          ;;BR IF YES
                                
027734  011203                        MOV     (R2),R3         ;GET THE WAS DATA
027736  104001                  5$:   ERROR   1               ;SUB DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 430      SUB TEST - SM0,DM1 - N:C = 0000
                                ;;*****************************************************************
027740                          TST430:
027740  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
027742  012700  000430                MOV     #430,R0         ;;LOAD R0 WITH TEST NUMBER
027746  013701  027774                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
027752  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
027756  012704  077777                MOV     #77777,R4       ;RESULT S / B = 77777
027762  012705  000001                MOV     #+1,R5          ;SRC OPR = +1
027766  012712  100000                MOV     #100000,(R2)    ;[DEST] = 100000
027772  000257                        CCC                     ;CLEAR FLAGS
                                
027774  160512                  2$:   SUB     R5,(R2)         ;TEST THE SUB
                                
027776  100403                        BMI     3$              ;N:C = 0010
030000  001402                        BEQ     3$
030002  102001                        BVC     3$
030004  103001                        BCC     4$
                                
030006  104001                  3$:   ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
                                
030010  020412                  4$:   CMP     R4,(R2)         ;CORRECT RESULT ?
030012  001402                        BEQ     TST431          ;;BR IF YES
                                
030014  011203                        MOV     (R2),R3         ;GET THE WAS DATA
```

```
7888  030016  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
7889
7890                                  ;;*****************************************************************
7891                                  ;*TEST 431       SUB TEST - SM1,DM0 - N:C = 0110
7892                                  ;;*****************************************************************
7893  030020                          TST431:
7894  030020  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7895  030022  012700  000431                 MOV     #431,R0         ;;LOAD R0 WITH TEST NUMBER
7896  030026  013701  030050                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7897  030032  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
7898  030036  012705  063760                 MOV     #DWTB+2,R5      ;SRC ADDR = DWTB+2
7899  030042  005003                         CLR     R3              ;[DEST] = 000000
7900  030044  000257                         CCC                     ;CLEAR FLAGS
7901  030046  000266                         266                     ;N:C = 0110
7902
7903  030050  161503                  2$:     SUB     (R5),R3         ;TEST THE SUB
7904
7905  030052  100003                         BPL     3$              ;N:C = 1001
7906  030054  001402                         BEQ     3$
7907  030056  102401                         BVS     3$
7908  030060  103401                         BCS     4$
7909
7910  030062  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7911
7912  030064  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7913  030066  001401                         BEQ     TST432          ;;BR IF YES
7914
7915  030070  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
7916
7917                                  ;;*****************************************************************
7918                                  ;*TEST 432       SUB TEST - SM1,DM0 - N:C = 1010
7919                                  ;;*****************************************************************
7920  030072                          TST432:
7921  030072  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7922  030074  012700  000432                 MOV     #432,R0         ;;LOAD R0 WITH TEST NUMBER
7923  030100  013701  030120                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7924  030104  005004                         CLR     R4              ;RESULT S / B = 000000
7925  030106  012705  063250                 MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
7926  030112  011503                         MOV     (R5),R3         ;[DEST] = 177777
7927  030114  000257                         CCC                     ;CLEAR FLAGS
7928  030116  000272                         272                     ;N:C = 1010
7929
7930  030120  161503                  2$:     SUB     (R5),R3 ;TEST THE SUB
7931
7932  030122  100403                         BMI     3$              ;N:C = 0100
7933  030124  001002                         BNE     3$
7934  030126  102401                         BVS     3$
7935  030130  103001                         BCC     4$
7936
7937  030132  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
7938
7939  030134  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
7940  030136  001401                         BEQ     TST433          ;;BR IF YES
7941
7942  030140  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
7943
```

```
                                ;;*****************************************************************
                                ;*TEST 433       SUB TEST - SM1,DM0 - N:C = 0000
                                ;;*****************************************************************
030142                          TST433:
030142  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030144  012700  000433                  MOV     #433,R0         ;;LOAD R0 WITH TEST NUMBER
030150  013701  030176                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030154  012704  077777                  MOV     #77777,R4               ;RESULT S / B = 77777
030160  012705  063242                  MOV     #MBUF1,R5               ;SRC ADDR =MBUF1
030164  012703  100000                  MOV     #100000,R3      ;[DEST] = 100000
030170  012715  000001                  MOV     #+1,(R5)        ;SRC OPR = +1
030174  000257                          CCC                     ;CLEAR FLAGS

030176  161503                  2$:     SUB     (R5),R3         ;TEST THE SUB

030200  100403                          BMI     3$              ;N:C = 0010
030202  001402                          BEQ     3$
030204  102001                          BVC     3$
030206  103001                          BCC     4$

030210  104002                  3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY

030212  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
030214  001401                          BEQ     TST434          ;;BR IF YES

030216  104002                  5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 434       SUB  SM1,DM1 TEST - N:C = 0110
                                ;;*****************************************************************
030220                          TST434:
030220  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030222  012700  000434                  MOV     #434,R0         ;;LOAD R0 WITH TEST NUMBER
030226  013701  030260                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030232  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030236  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
030242  012705  063242                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
030246  012715  000001                  MOV     #+1,(R5)        ;[SOURCE] = 000001
030252  005012                          CLR     (R2)            ;[DEST] = 000000
030254  000257                          CCC                     ;CLEAR FLAGS
030256  000266                          266                     ;N:C = 0110

030260  161512                  2$:     SUB     (R5),(R2)       ;TEST THE SUB

030262  100003                          BPL     3$              ;N:C = 1001 ?
030264  001402                          BEQ     3$
030266  102401                          BVS     3$
030270  103401                          BCS     4$

030272  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY

030274  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
030276  001402                          BEQ     TST435          ;;BR IF YES

030300  011203                          MOV     (R2),R3         ;GET THE WAS DATA
030302  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
```

```
8000
8001                                    ;;*****************************************************************
8002                                    ;*TEST 435      SUB  SM1,DM2 TEST - N:C = 0110
8003                                    ;;*****************************************************************
8004  030304                            TST435:
8005  030304  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8006  030306  012700  000435                   MOV     #435,R0         ;;LOAD R0 WITH TEST NUMBER
8007  030312  013701  030346                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8008  030316  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8009  030322  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
8010  030326  012705  063242                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
8011  030332  012715  000001                   MOV     #+1,(R5)        ;[SOURCE] = 000001
8012  030336  005012                           CLR     (R2)            ;[DEST] = 000000
8013  030340  010203                           MOV     R2,R3           ;R3 GETS DEST ADDR
8014  030342  000257                           CCC                     ;CLEAR FLAGS
8015  030344  000266                           266                     ;N:C = 0110
8016
8017  030346  161523                  2$:      SUB     (R5),(R3)+      ;TEST THE SUB
8018
8019  030350  100003                           BPL     3$              ;N:C = 1001 ?
8020  030352  001402                           BEQ     3$
8021  030354  102401                           BVS     3$
8022  030356  103401                           BCS     4$
8023
8024  030360  104005                  3$:      ERROR   5               ;SUB FAILED TO ALTER CODES PROPERLY
8025
8026  030362  020412                  4$:      CMP     R4,(R2)         ;CORRECT RESULT ?
8027  030364  001402                           BEQ     TST436          ;;BR IF YES
8028
8029  030366  011203                           MOV     (R2),R3         ;GET THE WAS DATA
8030  030370  104001                  5$:      ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8031
8032                                    ;;*****************************************************************
8033                                    ;*TEST 436      NEG DM2 TEST
8034                                    ;;*****************************************************************
8035  030372                            TST436:
8036  030372  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8037  030374  012700  000436                   MOV     #436,R0         ;;LOAD R0 WITH TEST NUMBER
8038  030400  013701  030424                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8039  030404  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8040  030410  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
8041  030414  010205                           MOV     R2,R5           ;[R5] = DEST ADDR
8042  030416  012712  052526                   MOV     #52526,(R2)     ;[DEST] = 52526
8043  030422  000257                           CCC                     ;SCOPE SYNC
8044
8045  030424  005425                  2$:      NEG     (R5)+           ;TEST THE NEG - MODE 2
8046
8047  030426  020412                           CMP     R4,(R2)         ;RESULT = 125252?
8048  030430  001402                           BEQ     4$              ;BR IF YES
8049
8050  030432  011203                           MOV     (R2),R3         ;GET THE WAS DATA
8051  030434  104001                  3$:      ERROR   1               ;NEG DELIVERED WRONG RESULT
8052
8053  030436  022705  063240          4$:      CMP     #MBUF0+2,R5     ;DID REG. GET AUTO INCREMENTED?
8054  030442  001401                           BEQ     TST437          ;;BR IF YES
8055
```

```
8056  030444  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8057
8058                                  ;;*****************************************************************
8059                                  ;*TEST 437       NEG DM3 TEST
8060                                  ;;*****************************************************************
8061  030446                          TST437:
8062  030446  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8063  030450  012700  000437                 MOV     #437,R0         ;;LOAD R0 WITH TEST NUMBER
8064  030454  013701  030502                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8065  030460  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8066  030464  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
8067  030470  012705  063232                 MOV     #ATA+10,R5      ;[ATA+10] = MBUF0
8068  030474  012712  052526                 MOV     #52526,(R2)     ;[DEST] = 52526
8069  030500  000257                         CCC                     ;SCOPE SYNC
8070
8071  030502  005435                  2$:     NEG     @(R5)+          ;TEST THE NEG - MODE 3
8072
8073  030504  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8074  030506  001402                          BEQ     4$              ;BR IF YES
8075
8076  030510  011203                          MOV     (R2),R3         ;GET WAS DATA
8077  030512  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8078
8079  030514  022705  063234          4$:     CMP     #ATA+12,R5      ;DID REG GET AUTO INCREMENTED?
8080  030520  001401                          BEQ     TST440          ;;BR IF YES
8081
8082  030522  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8083
8084                                  ;;*****************************************************************
8085                                  ;*TEST 440       NEG DM4 TEST
8086                                  ;;*****************************************************************
8087  030524                          TST440:
8088  030524  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8089  030526  012700  000440                 MOV     #440,R0         ;;LOAD R0 WITH TEST NUMBER
8090  030532  013701  030560                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8091  030536  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8092  030542  012704  125252                 MOV     #125252,R4      ;RESULT S / B = 125252
8093  030546  012705  063240                 MOV     #MBUF0+2,R5     ;[R5] = DEST ADDR + 2
8094  030552  012712  052526                 MOV     #52526,(R2)     ;[DEST] = 52526
8095  030556  000257                         CCC                     ;SCOPE SYNC
8096
8097  030560  005445                  2$:     NEG     -(R5)           ;TEST THE NEG - MODE 4
8098
8099  030562  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8100  030564  001402                          BEQ     4$              ;BR IF YES
8101
8102  030566  011203                          MOV     (R2),R3         ;GET WAS DATA
8103  030570  104001                  3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8104
8105  030572  020502                  4$:     CMP     R5,R2           ;DID REG GET AUTO INCREMENTED?
8106  030574  001401                          BEQ     TST441          ;;BR IF YES
8107
8108  030576  104005                  5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
8109
8110                                  ;;*****************************************************************
8111                                  ;*TEST 441       NEG DM5 TEST
```

```
8112                                    ;;*****************************************************************
8113  030600                            TST441:
8114  030600  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8115  030602  012700  000441                    MOV     #441,R0         ;;LOAD R0 WITH TEST NUMBER
8116  030606  013701  030634                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8117  030612  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8118  030616  012704  125252                    MOV     #125252,R4      ;RESULT S / B = 125252
8119  030622  012705  063234                    MOV     #ATA+12,R5      ;[R5] = (ADR OF MBUF0) +2
8120  030626  012712  052526                    MOV     #52526,(R2)     ;[DEST] = 52526
8121  030632  000257                            CCC                     ;SCOPE SYNC
8122
8123  030634  005455                    2$:     NEG     @-(R5)          ;TEST THE NEG - MODE 5
8124
8125  030636  020412                            CMP     R4,(R2)         ;RESULT = 125252?
8126  030640  001402                            BEQ     4$              ;BR IF YES
8127
8128  030642  011203                            MOV     (R2),R3         ;GET WAS DATA
8129  030644  104001                    3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8130
8131  030646  022705  063232            4$:     CMP     #ATA+10,R5      ;DID NEG UPDATE REG
8132  030652  001401                            BEQ     TST442          ;;BR IF YES
8133
8134  030654  104005                    5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
8135
8136                                    ;;*****************************************************************
8137                                    ;*TEST 442       NEG DM6 TEST
8138                                    ;;*****************************************************************
8139  030656                            TST442:
8140  030656  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8141  030660  012700  000442                    MOV     #442,R0         ;;LOAD R0 WITH TEST NUMBER
8142  030664  013701  030712                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8143  030670  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8144  030674  012704  125252                    MOV     #125252,R4      ;RESULT S / B = 125252
8145  030700  012705  063234                    MOV     #MBUF0-2,R5     ;[R5] = BASE ADDR
8146  030704  012712  052526                    MOV     #52526,(R2)     ;[DEST] = 52526
8147  030710  000257                            CCC                     ;SCOPE SYNC
8148
8149  030712  005465  000002            2$:     NEG     2(R5)           ;TEST THE NEG - MODE 6
8150
8151  030716  020412                            CMP     R4,(R2)         ;RESULT = 125252?
8152  030720  001402                            BEQ     TST443          ;;BR IF YES
8153
8154  030722  011203                            MOV     (R2),R3         ;GET WAS DATA
8155  030724  104001                    3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8156
8157                                    ;;*****************************************************************
8158                                    ;*TEST 443       NEG DM7 TEST
8159                                    ;;*****************************************************************
8160  030726                            TST443:
8161  030726  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8162  030730  012700  000443                    MOV     #443,R0         ;;LOAD R0 WITH TEST NUMBER
8163  030734  013701  030762                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8164  030740  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8165  030744  012704  125252                    MOV     #125252,R4      ;RESULT S / B = 125252
8166  030750  012705  063222                    MOV     #ATA,R5         ;[R5] = BASE ADDR
8167  030754  012712  052526                    MOV     #52526,(R2)     ;[DEST] = 52526
```

```
8168  030760  000257                          CCC                     ;SCOPE SYNC
8169
8170  030762  005475  000010        2$:       NEG     @10(R5)         ;TEST THE NEG - MODE 7
8171
8172  030766  020412                          CMP     R4,(R2)         ;RESULT = 125252?
8173  030770  001402                          BEQ     TST444          ;;BR IF YES
8174
8175  030772  011203                          MOV     (R2),R3         ;GET WAS DATA
8176  030774  104001                3$:       ERROR   1               ;NEG DELIVERED WRONG RESULT
8177
8178                                          ;;**************************************************************
8179                                          ;*TEST 444      MOV SM1,DM1 TEST - N:C = 0100
8180                                          ;;**************************************************************
8181  030776                                  TST444:
8182  030776  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8183  031000  012700  000444                  MOV     #444,R0         ;;LOAD R0 WITH TEST NUMBER
8184  031004  013701  031034                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8185  031010  005004                          CLR     R4              ;RESULT S / B = 177777
8186  031012  005104                          COM     R4
8187  031014  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8188  031020  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8189  031024  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8190  031026  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8191  031030  000257                          CCC                     ;CLEAR FLAGS
8192  031032  000264                          264                     ;N:C = 0100
8193
8194  031034  011513                2$:       MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8195
8196  031036  100003                          BPL     3$              ;N:C = 1000 ?
8197  031040  001402                          BEQ     3$
8198  031042  102401                          BVS     3$
8199  031044  103001                          BCC     4$
8200
8201  031046  104001                3$:       ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8202
8203  031050  020412                4$:       CMP     R4,(R2)         ;RESULT CORRECT ??
8204  031052  001403                          BEQ     TST445          ;;BR IF YES
8205
8206  031054  005003                          CLR     R3              ;GET THE WAS DATA
8207  031056  051203                          BIS     (R2),R3
8208  031060  104001                5$:       ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8209
8210                                          ;;**************************************************************
8211                                          ;*TEST 445      MOV SM2,DM1 TEST - N:C = 0100
8212                                          ;;**************************************************************
8213  031062                                  TST445:
8214  031062  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8215  031064  012700  000445                  MOV     #445,R0         ;;LOAD R0 WITH TEST NUMBER
8216  031070  013701  031120                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8217  031074  005004                          CLR     R4              ;RESULT S / B = 177777
8218  031076  005104                          COM     R4
8219  031100  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8220  031104  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8221  031110  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8222  031112  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8223  031114  000257                          CCC                     ;CLEAR FLAGS
```

```
8224  031116  000264                          264                     ;N:C = 0100
8225
8226  031120  012513                  2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
8227
8228  031122  100003                          BPL     3$              ;N:C = 1000 ?
8229  031124  001402                          BEQ     3$
8230  031126  102401                          BVS     3$
8231  031130  103001                          BCC     4$
8232
8233  031132  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8234
8235  031134  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8236  031136  001403                          BEQ     TST446          ;;BR IF YES
8237
8238  031140  005003                          CLR     R3              ;GET THE WAS DATA
8239  031142  051203                          BIS     (R2),R3
8240  031144  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8241
8242                                  ;;******************************************************************
8243                                  ;*TEST 446       MOV SM1,DM1 TEST - N:C = 1011
8244                                  ;;******************************************************************
8245  031146                          TST446:
8246  031146  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8247  031150  012700  000446                  MOV     #446,R0         ;;LOAD R0 WITH TEST NUMBER
8248  031154  013701  031204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8249  031160  005004                          CLR     R4              ;RESULT S / B = 000000
8250  031162  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8251  031166  012705  063246                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
8252  031172  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8253  031174  005012                          CLR     (R2)            ;MAKE [DEST] = 177777
8254  031176  005112                          COM     (R2)
8255  031200  000257                          CCC                     ;CLEAR FLAGS
8256  031202  000273                          273                     ;N:C = 1011
8257
8258  031204  011513                  2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8259
8260  031206  100403                          BMI     3$              ;N:C = 0101 ?
8261  031210  001002                          BNE     3$
8262  031212  102401                          BVS     3$
8263  031214  103401                          BCS     4$
8264
8265  031216  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8266
8267  031220  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8268  031222  001403                          BEQ     TST447          ;;BR IF YES
8269
8270  031224  005003                          CLR     R3              ;GET THE WAS DATA
8271  031226  051203                          BIS     (R2),R3
8272  031230  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8273
8274                                  ;;******************************************************************
8275                                  ;*TEST 447       MOV SM2,DM1 TEST - N:C = 1011
8276                                  ;;******************************************************************
8277  031232                          TST447:
8278  031232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8279  031234  012700  000447                  MOV     #447,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
8280  031240  013701  031270          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8281  031244  005004                  CLR     R4              ;RESULT S / B = 000000
8282  031246  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8283  031252  012705  063246          MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
8284  031256  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8285  031260  005012                  CLR     (R2)            ;MAKE [DEST] = 177777
8286  031262  005112                  COM     (R2)
8287  031264  000257                  CCC                     ;CLEAR FLAGS
8288  031266  000273                  273                     ;N:C = 1011
8289
8290  031270  012513          2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
8291
8292  031272  100403                  BMI     3$              ;N:C = 0101 ?
8293  031274  001002                  BNE     3$
8294  031276  102401                  BVS     3$
8295  031300  103401                  BCS     4$
8296
8297  031302  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8298
8299  031304  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8300  031306  001403                  BEQ     TST450          ;;BR IF YES
8301
8302  031310  005003                  CLR     R3              ;GET THE WAS DATA
8303  031312  051203                  BIS     (R2),R3
8304  031314  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8305
8306                          ;;**************************************************************
8307                          ;*TEST 450      MOV SM1,DM2 TEST - N:C = 0100
8308                          ;;**************************************************************
8309  031316                  TST450:
8310  031316  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8311  031320  012700  000450          MOV     #450,R0         ;;LOAD R0 WITH TEST NUMBER
8312  031324  013701  031354          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8313  031330  005004                  CLR     R4              ;RESULT S / B = 177777
8314  031332  005104                  COM     R4
8315  031334  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8316  031340  012705  063250          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
8317  031344  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8318  031346  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8319  031350  000257                  CCC                     ;CLEAR FLAGS
8320  031352  000264                  264                     ;N:C = 0100
8321
8322  031354  011523          2$:     MOV     (R5),(R3)+      ;TEST THE MOV - SM1,DM2
8323
8324  031356  100003                  BPL     3$              ;N:C =  1000 ?
8325  031360  001402                  BEQ     3$
8326  031362  102401                  BVS     3$
8327  031364  103001                  BCC     4$
8328
8329  031366  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8330
8331  031370  022703  063240          CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
8332  031374  001401                  BEQ     6$              ;BR IF YES
8333
8334  031376  104005          5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8335
```

```
8336  031400  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8337  031402  001403                          BEQ     TST451          ;;BR IF YES
8338
8339  031404  005003                          CLR     R3              ;GET THE WAS DATA
8340  031406  051203                          BIS     (R2),R3
8341  031410  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8342
8343                                  ;;**********************************************************************
8344                                  ;*TEST 451       MOV SM2,DM2 TEST - N:C = 0100
8345                                  ;;**********************************************************************
8346  031412                          TST451:
8347  031412  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8348  031414  012700  000451                  MOV     #451,R0         ;;LOAD R0 WITH TEST NUMBER
8349  031420  013701  031450                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8350  031424  005004                          CLR     R4              ;RESULT S / B = 177777
8351  031426  005104                          COM     R4
8352  031430  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8353  031434  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
8354  031440  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8355  031442  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8356  031444  000257                          CCC                     ;CLEAR FLAGS
8357  031446  000264                          264                     ;N:C = 0100
8358
8359  031450  012523                  2$:     MOV     (R5)+,(R3)+     ;TEST THE MOV - SM2,DM2
8360
8361  031452  100003                          BPL     3$              ;N:C = 1000 ?
8362  031454  001402                          BEQ     3$
8363  031456  102401                          BVS     3$
8364  031460  103001                          BCC     4$
8365
8366  031462  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8367
8368  031464  022703  063240          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
8369  031470  001401                          BEQ     6$              ;BR IF YES
8370
8371  031472  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8372
8373  031474  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8374  031476  001403                          BEQ     TST452          ;;BR IF YES
8375
8376  031500  005003                          CLR     R3              ;GET THE WAS DATA
8377  031502  051203                          BIS     (R2),R3
8378  031504  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8379
8380                                  ;;**********************************************************************
8381                                  ;*TEST 452       MOV SM1,DM3 TEST - N:C = 0100
8382                                  ;;**********************************************************************
8383  031506                          TST452:
8384  031506  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8385  031510  012700  000452                  MOV     #452,R0         ;;LOAD R0 WITH TEST NUMBER
8386  031514  013701  031546                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8387  031520  005004                          CLR     R4              ;RESULT S / B = 177777
8388  031522  005104                          COM     R4
8389  031524  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8390  031530  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8391  031534  012703  063232                  MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
```

```
8392  031540  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
8393  031542  000257                    CCC                     ;CLEAR FLAGS
8394  031544  000264                    264                     ;N:C = 0100
8395
8396  031546  011533            2$:     MOV     (R5),@(R3)+     ;TEST THE MOV - SM1,DM3
8397
8398  031550  100003                    BPL     3$              ;N:C = 1000 ?
8399  031552  001402                    BEQ     3$
8400  031554  102401                    BVS     3$
8401  031556  103001                    BCC     4$
8402
8403  031560  104001            3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8404
8405  031562  022703  063234    4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
8406  031566  001401                    BEQ     6$              ;BR IF YES
8407
8408  031570  104005            5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8409
8410  031572  020412            6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8411  031574  001403                    BEQ     TST453          ;;BR IF YES
8412
8413  031576  005003                    CLR     R3              ;GET THE WAS DATA
8414  031600  051203                    BIS     (R2),R3
8415  031602  104001            7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8416
8417                            ;;**************************************************************
8418                            ;*TEST 453      MOV SM2,DM3 TEST - N:C = 0100
8419                            ;;**************************************************************
8420  031604                    TST453:
8421  031604  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8422  031606  012700  000453            MOV     #453,R0         ;;LOAD R0 WITH TEST NUMBER
8423  031612  013701  031644            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8424  031616  005004                    CLR     R4              ;RESULT S / B = 177777
8425  031620  005104                    COM     R4
8426  031622  012702  063236            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8427  031626  012705  063250            MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8428  031632  012703  063232            MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
8429  031636  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
8430  031640  000257                    CCC                     ;CLEAR FLAGS
8431  031642  000264                    264                     ;N:C = 0100
8432
8433  031644  012533            2$:     MOV     (R5)+,@(R3)+    ;TEST THE MOV - SM2,DM3
8434
8435  031646  100003                    BPL     3$              ;N:C = 1000 ?
8436  031650  001402                    BEQ     3$
8437  031652  102401                    BVS     3$
8438  031654  103001                    BCC     4$
8439
8440  031656  104001            3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8441
8442  031660  022703  063234    4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
8443  031664  001401                    BEQ     6$              ;BR IF YES
8444
8445  031666  104005            5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8446
8447  031670  020412            6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
```

```
8448  031672  001403                          BEQ     TST454          ;;BR IF YES
8449
8450  031674  005003                          CLR     R3              ;GET THE WAS DATA
8451  031676  051203                          BIS     (R2),R3
8452  031700  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8453
8454                                  ;;**********************************************************************
8455                                  ;*TEST 454       MOV SM1,DM4 TEST - N:C = 0100
8456                                  ;;**********************************************************************
8457  031702                          TST454:
8458  031702  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8459  031704  012700  000454                  MOV     #454,R0         ;;LOAD R0 WITH TEST NUMBER
8460  031710  013701  031742                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8461  031714  005004                          CLR     R4              ;RESULT S / B = 177777
8462  031716  005104                          COM     R4
8463  031720  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8464  031724  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8465  031730  012703  063240                  MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
8466  031734  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8467  031736  000257                          CCC                     ;CLEAR FLAGS
8468  031740  000264                          264                     ;N:C = 0100
8469
8470  031742  011543                  2$:     MOV     (R5),-(R3)      ;TEST THE MOV - SM1,DM4
8471
8472  031744  100003                          BPL     3$              ;N:C = 1000 ?
8473  031746  001402                          BEQ     3$
8474  031750  102401                          BVS     3$
8475  031752  103001                          BCC     4$
8476
8477  031754  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8478
8479  031756  020203                  4$:     CMP     R2,R3           ;DID MOV DECREMENT DEST REG ?
8480  031760  001401                          BEQ     6$              ;BR IF YES
8481
8482  031762  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8483
8484  031764  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8485  031766  001403                          BEQ     TST455          ;;BR IF YES
8486
8487  031770  005003                          CLR     R3              ;GET THE WAS DATA
8488  031772  051203                          BIS     (R2),R3
8489  031774  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8490
8491                                  ;;**********************************************************************
8492                                  ;*TEST 455       MOV SM2,DM4 TEST - N:C = 0100
8493                                  ;;**********************************************************************
8494  031776                          TST455:
8495  031776  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8496  032000  012700  000455                  MOV     #455,R0         ;;LOAD R0 WITH TEST NUMBER
8497  032004  013701  032036                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8498  032010  005004                          CLR     R4              ;RESULT S / B = 177777
8499  032012  005104                          COM     R4
8500  032014  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8501  032020  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8502  032024  012703  063240                  MOV     #MBUF0+2,R3             ;BASE DEST ADDR = MBUF0+2
8503  032030  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
```

```
8504  032032  000257                          CCC                     ;CLEAR FLAGS
8505  032034  000264                          264                     ;N:C = 0100
8506
8507  032036  012543              2$:         MOV     (R5)+,-(R3)     ;TEST THE MOV - SM2,DM4
8508
8509  032040  100003                          BPL     3$              ;N:C = 1000 ?
8510  032042  001402                          BEQ     3$
8511  032044  102401                          BVS     3$
8512  032046  103001                          BCC     4$
8513
8514  032050  104001              3$:         ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8515
8516  032052  020203              4$:         CMP     R2,R3           ;DID MOV INCREMENT DEST REG ?
8517  032054  001401                          BEQ     6$              ;BR IF YES
8518
8519  032056  104005              5$:         ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8520
8521  032060  020412              6$:         CMP     R4,(R2)         ;RESULT CORRECT ??
8522  032062  001403                          BEQ     TST456          ;;BR IF YES
8523
8524  032064  005003                          CLR     R3              ;GET THE WAS DATA
8525  032066  051203                          BIS     (R2),R3
8526  032070  104001              7$:         ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8527
8528                              ;;**************************************************************
8529                              ;*TEST 456       MOV SM1,DM5 TEST - N:C = 0100
8530                              ;;**************************************************************
8531  032072                      TST456:
8532  032072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8533  032074  012700  000456                  MOV     #456,R0         ;;LOAD R0 WITH TEST NUMBER
8534  032100  013701  032144                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8535                              .SBTTL USER CONTROLLED BREAKPOINT -- BIT9
8536  032104  032737  001000  063160          BIT     #BIT9,@#BPTLOC  ;BREAKPOINT HALT SET ??
8537  032112  001401                          BEQ     .+4             ;BR IF NOT
8538  032114  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
8539  032116  005004                          CLR     R4              ;RESULT S / B = 177777
8540  032120  005104                          COM     R4
8541  032122  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8542  032126  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8543  032132  012703  063234                  MOV     #ATA+12,R3              ;BASE DEST ADDR = ATA+12
8544  032136  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
8545  032140  000257                          CCC                     ;CLEAR FLAGS
8546  032142  000264                          264                     ;N:C = 0100
8547
8548  032144  011553              2$:         MOV     (R5),@-(R3)     ;TEST THE MOV - SM1,DM5
8549
8550  032146  100003                          BPL     3$              ;N:C = 0100 ?
8551  032150  001402                          BEQ     3$
8552  032152  102401                          BVS     3$
8553  032154  103001                          BCC     4$
8554
8555  032156  104001              3$:         ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8556
8557  032160  022703  063232      4$:         CMP     #ATA+10.R3      ;DID MOV DECREMENT DEST REG ?
8558  032164  001401                          BEQ     6$              ;BR IF YES
8559
```

```
8560  032166  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8561
8562  032170  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8563  032172  001403                          BEQ     TST457          ;;BR IF YES
8564
8565  032174  005003                          CLR     R3              ;GET THE WAS DATA
8566  032176  051203                          BIS     (R2),R3
8567  032200  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8568
8569                                  ;;**************************************************************
8570                                  ;*TEST 457       MOV SM2,DM5 TEST - N:C = 0100
8571                                  ;;**************************************************************
8572  032202                          TST457:
8573  032202  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8574  032204  012700  000457                  MOV     #457,R0         ;;LOAD R0 WITH TEST NUMBER
8575  032210  013701  032242                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8576  032214  005004                          CLR     R4              ;RESULT S / B = 177777
8577  032216  005104                          COM     R4
8578  032220  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8579  032224  012705  063250                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8580  032230  012703  063234                  MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
8581  032234  005012                          CLR     (R2)            ;MAKE [DEST] = 00000
8582  032236  000257                          CCC                     ;CLEAR FLAGS
8583  032240  000264                          264                     ;N:C = 1000
8584
8585  032242  012553                  2$:     MOV     (R5)+,@-(R3)    ;TEST THE MOV - SM2,DM5
8586
8587  032244  100003                          BPL     3$              ;N:C = 1000 ?
8588  032246  001402                          BEQ     3$
8589  032250  102401                          BVS     3$
8590  032252  103001                          BCC     4$
8591
8592  032254  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8593
8594  032256  022703  063232          4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
8595  032262  001401                          BEQ     6$              ;BR IF YES
8596
8597  032264  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
8598
8599  032266  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8600  032270  001403                          BEQ     TST460          ;;BR IF YES
8601
8602  032272  005003                          CLR     R3              ;GET THE WAS DATA
8603  032274  051203                          BIS     (R2),R3
8604  032276  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8605
8606                                  ;;**************************************************************
8607                                  ;*TEST 460       MOV SM1,DM6 TEST - N:C = 0100
8608                                  ;;**************************************************************
8609  032300                          TST460:
8610  032300  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8611  032302  012700  000460                  MOV     #460,R0         ;;LOAD R0 WITH TEST NUMBER
8612  032306  013701  032340                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8613  032312  005004                          CLR     R4              ;RESULT S / B = 177777
8614  032314  005104                          COM     R4
8615  032316  012702  063244                  MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
```

```
8616  032322  012705  063250          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8617  032326  012703  063236          MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
8618  032332  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8619  032334  000257                  CCC                     ;CLEAR FLAGS
8620  032336  000264                  264                     ;N:C = 0100
8621
8622  032340  011563  000006  2$:     MOV     (R5),6(R3)      ;TEST THE MOV - SM1,DM6
8623
8624  032344  100003                  BPL     3$              ;N:C = 1000 ?
8625  032346  001402                  BEQ     3$
8626  032350  102401                  BVS     3$
8627  032352  103001                  BCC     4$
8628
8629  032354  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8630
8631  032356  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8632  032360  001403                  BEQ     TST461          ;;BR IF YES
6633
8634  032362  005003                  CLR     R3              ;GET THE WAS DATA
8635  032364  051203                  BIS     (R2),R3
8636  032366  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8637
8638                          ;;**********************************************************************
8639                          ;*TEST 461       MOV SM2,DM6 TEST - N:C = 0100
8640                          ;;**********************************************************************
8641  032370                  TST461:
8642  032370  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8643  032372  012700  000461          MOV     #461,R0         ;;LOAD R0 WITH TEST NUMBER
8644  032376  013701  032430          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8645  032402  005004                  CLR     R4              ;RESULT S / B = 177777
8646  032404  005104                  COM     R4
8647  032406  012702  063244          MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
8648  032412  012705  063250          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8649  032416  012703  063236          MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
8650  032422  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8651  032424  000257                  CCC                     ;CLEAR FLAGS
8652  032426  000264                  264                     ;N:C = 0100
8653
8654  032430  012563  000006  2$:     MOV     (R5)+,6(R3)     ;TEST THE MOV - SM2,DM6
8655
8656  032434  100003                  BPL     3$              ;N:C = 1000 ?
8657  032436  001402                  BEQ     3$
8658  032440  102401                  BVS     3$
8659  032442  103001                  BCC     4$
8660
8661  032444  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8662
8663  032446  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8664  032450  001403                  BEQ     TST462          ;;BR IF YES
8665
8666  032452  005003                  CLR     R3              ;GET THE WAS DATA
8667  032454  051203                  BIS     (R2),R3
8668  032456  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8669
8670                          ;;**********************************************************************
8671                          ;*TEST 462       MOV SM1,DM7 TEST - N:C = 0100
```

```
8672                                    ;;*****************************************************************
8673  032460                            TST462:
8674  032460  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8675  032462  012700  000462                    MOV     #462,R0         ;;LOAD R0 WITH TEST NUMBER
8676  032466  013701  032520                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8677  032472  005004                            CLR     R4              ;RESULT S / B = 177777
8678  032474  005104                            COM     R4
8679  032476  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8680  032502  012705  063250                    MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8681  032506  012703  063222                    MOV     #ATA,R3         ;BASE DEST ADDR = ATA
8682  032512  005012                            CLR     (R2)            ;MAKE [DEST] = 000000
8683  032514  000257                            CCC                     ;CLEAR FLAGS
8684  032516  000264                            264                     ;N:C = 0100
8685
8686  032520  011573  000010           2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM1,DM7
8687
8688  032524  100003                            BPL     3$              ;N:C = 1000 ?
8689  032526  001402                            BEQ     3$
8690  032530  102401                            BVS     3$
8691  032532  103001                            BCC     4$
8692
8693  032534  104001                    3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8694
8695  032536  020412                    4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8696  032540  001403                            BEQ     TST463          ;;BR IF YES
8697
8698  032542  005003                            CLR     R3              ;GET THE WAS DATA
8699  032544  051203                            BIS     (R2),R3
8700  032546  104001                    5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8701
8702                                    ;;*****************************************************************
8703                                    ;*TEST 463      MOV SM2,DM7 TEST - N:C = 0100
8704                                    ;;*****************************************************************
8705  032550                            TST463:
8706  032550  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8707  032552  012700  000463                    MOV     #463,R0         ;;LOAD R0 WITH TEST NUMBER
8708  032556  013701  032610                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8709  032562  005004                            CLR     R4              ;RESULT S / B = 177777
8710  032564  005104                            COM     R4
8711  032566  012702  063236                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8712  032572  012705  063250                    MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8713  032576  012703  063222                    MOV     #ATA,R3         ;BASE DEST ADDR = ATA
8714  032602  005012                            CLR     (R2)            ;MAKE [DEST] = 000000
8715  032604  000257                            CCC                     ;CLEAR FLAGS
8716  032606  000264                            264                     ;N:C = 0100
8717
8718  032610  011573  000010           2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM2,DM7
8719
8720  032614  100003                            BPL     3$              ;N:C = 1000 ?
8721  032616  001402                            BEQ     3$
8722  032620  102401                            BVS     3$
8723  032622  103001                            BCC     4$
8724
8725  032624  104001                    3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8726
8727  032626  020412                    4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
```

```
8728  032630  001403                          BEQ     TST464          ;;BR IF YES
8729
8730  032632  005003                          CLR     R3              ;GET THE WAS DATA
8731  032634  051203                          BIS     (R2),R3
8732  032636  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8733
8734                                  ;;*****************************************************************
8735                                  ;*TEST 464       MOV  SM0,DM1 TEST
8736                                  ;;*****************************************************************
8737  032640                          TST464:
8738  032640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8739  032642  012700  000464                  MOV     #464,R0         ;;LOAD R0 WITH TEST NUMBER
8740  032646  013701  032666                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8741  032652  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8742  032656  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
8743  032660  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
8744  032662  005012                          CLR     (R2)            ;[DEST] = 000000
8745  032664  000257                          CCC                     ;SCOPE SYNC
8746
8747  032666  010015                  2$:     MOV     R0,(R5)         ;TEST THE MOV
8748
8749  032670  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
8750  032672  001402                          BEQ     TST465          ;;BR IF YES
8751
8752  032674  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8753  032676  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8754
8755                                  ;;*****************************************************************
8756                                  ;*TEST 465       MOV  SM0,DM2 TEST
8757                                  ;;*****************************************************************
8758  032700                          TST465:
8759  032700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8760  032702  012700  000465                  MOV     #465,R0         ;;LOAD R0 WITH TEST NUMBER
8761  032706  013701  032726                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8762  032712  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8763  032716  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
8764  032720  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
8765  032722  005012                          CLR     (R2)            ;[DEST] = 000000
8766  032724  000257                          CCC                     ;SCOPE SYNC
8767
8768  032726  010025                  2$:     MOV     R0,(R5)+        ;TEST THE MOV
8769
8770  032730  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
8771  032732  001402                          BEQ     TST466          ;;BR IF YES
8772
8773  032734  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8774  032736  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8775
8776                                  ;;*****************************************************************
8777                                  ;*TEST 466       MOV  SM0,DM3 TEST
8778                                  ;;*****************************************************************
8779  032740                          TST466:
8780  032740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8781  032742  012700  000466                  MOV     #466,R0         ;;LOAD R0 WITH TEST NUMBER
8782  032746  013701  032770                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8783  032752  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
8784  032756  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
8785  032760  012705  063232                MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
8786  032764  005012                        CLR     (R2)            ;[DEST] = 000000
8787  032766  000257                        CCC                     ;SCOPE SYNC
8788
8789  032770  010035                2$:     MOV     R0,@(R5)+       ;TEST THE MOV
8790
8791  032772  020412                        CMP     R4,(R2)         ;CORRECT RESULT
8792  032774  001402                        BEQ     TST467          ;;BR IF YES
8793
8794  032776  011203                        MOV     (R2),R3         ;GET THE WAS DATA
8795  033000  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8796
8797                                ;;**********************************************************************
8798                                ;*TEST 467      MOV  SM0,DM4 TEST
8799                                ;;**********************************************************************
8800  033002                        TST467:
8801  033002  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8802  033004  012700  000467                MOV     #467,R0         ;;LOAD R0 WITH TEST NUMBER
8803  033010  013701  033032                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8804  033014  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8805  033020  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
8806  033022  012705  063240                MOV     #MBUF0+2,R5     ;R5 CONTAINS BASE DEST ADDR
8807  033026  005012                        CLR     (R2)            ;[DEST] = 000000
8808  033030  000257                        CCC                     ;SCOPE SYNC
8809
8810  033032  010045                2$:     MOV     R0,-(R5)        ;TEST THE MOV
8811
8812  033034  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
8813  033036  001402                        BEQ     TST470          ;;BR IF YES
8814
8815  033040  011203                        MOV     (R2),R3         ;GET THE WAS DATA
8816  033042  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8817
8818                                ;;**********************************************************************
8819                                ;*TEST 470      MOV  SM0,DM5 TEST
8820                                ;;**********************************************************************
8821  033044                        TST470:
8822  033044  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8823  033046  012700  000470                MOV     #470,R0         ;;LOAD R0 WITH TEST NUMBER
8824  033052  013701  033074                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8825  033056  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8826  033062  010004                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
8827  033064  012705  063234                MOV     #ATA+12,R5      ;R5 CONTAINS BASE DEST ADDR
8828  033070  005012                        CLR     (R2)            ;[DEST] = 000000
8829  033072  000257                        CCC                     ;SCOPE SYNC
8830
8831  033074  010055                2$:     MOV     R0,@-(R5)       ;TEST THE MOV
8832
8833  033076  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
8834  033100  001402                        BEQ     TST471          ;;BR IF YES
8835
8836  033102  011203                        MOV     (R2),R3         ;GET THE WAS DATA
8837  033104  104001                3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8838
8839
```

```
                                ;;*****************************************************************
                                ;*TEST 471       MOV  SMO,DM6 TEST
                                ;;*****************************************************************
                                TST471:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #471,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
                                        MOV     R0,R4           ;RESULT S / B = TEST NUMBER
                                        MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
                                        CLR     (R2)            ;[DEST] = 000000
                                        CCC                     ;SCOPE SYNC

                                2$:     MOV     R0,4(R5)        ;TEST THE MOV

                                        CMP     R4,(R2)         ;RESULT CORRECT ?
                                        BEQ     TST472          ;;BR IF YES

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

                                ;;*****************************************************************
                                ;*TEST 472       MOVB TEST - SMO,DMO - EXTEND 1'S
                                ;;*****************************************************************
                                TST472:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #472,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177652,R4      ;RESULT S / B = 177652
                                        MOV     #252,R5         ;SOURCE OP = 252
                                        CLR     R3              ;[DEST] = 000000
                                        CCC                     ;CLEAR FLAGS
                                        266                     ;N:C = 0110

                                2$:     MOVB    R5,R3           ;TEST THE MOVB

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

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

                                4$:     CMP     R4,R3           ;RESULT CORRECT ?
                                        BEQ     TST473          ;;BR IF YES

                                5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 473       MOVB TEST - SMO,DMO - EXTEND 0'S
                                ;;*****************************************************************
                                TST473:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #473,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        CLR     R4              ;RESULT S / B = 000000
```

```
8896  033240  012705  177400              MOV     #177400,R5      ;SOURCE OP = 177400
8897  033244  005003                      CLR     R3              ;[DEST] = 177777
8898  033246  005103                      COM     R3
8899  033250  000257                      CCC                     ;CLEAR FLAGS
8900  033252  000271                      271                     ;N:C = 1001
8901
8902  033254  110503              2$:     MOVB    R5,R3           ;TEST THE MOVB
8903
8904  033256  100403                      BMI     3$              ;N:C = 0101 ?
8905  033260  001002                      BNE     3$
8906  033262  102401                      BVS     3$
8907  033264  103401                      BCS     4$
8908
8909  033266  104002              3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
8910
8911  033270  020403              4$:     CMP     R4,R3           ;RESULT CORRECT ?
8912  033272  001401                      BEQ     TST474          ;;BR IF YES
8913
8914  033274  104002              5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
8915
8916                              ;;**************************************************************
8917                              ;*TEST 474       MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
8918                              ;;**************************************************************
8919  033276                      TST474:
8920  033276  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8921  033300  012700  000474              MOV     #474,R0         ;;LOAD R0 WITH TEST NUMBER
8922  033304  013701  033324              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8923  033310  005004                      CLR     R4              ;RESULT S / B = 000000
8924  033312  012705  064554              MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
8925  033316  005003                      CLR     R3              ;[DEST] = 177777
8926  033320  005103                      COM     R3
8927  033322  000257                      CCC                     ;SCOPE SYNC
8928
8929  033324  111503              2$:     MOVB    (R5),R3         ;TEST THE MOVB
8930
8931  033326  020403                      CMP     R4,R3           ;RESULT CORRECT ?
8932  033330  001401                      BEQ     TST475          ;;BR IF YES
8933
8934  033332  104002              3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
8935
8936                              ;;**************************************************************
8937                              ;*TEST 475       MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
8938                              ;;**************************************************************
8939  033334                      TST475:
8940  033334  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8941  033336  012700  000475              MOV     #475,R0         ;;LOAD R0 WITH TEST NUMBER
8942  033342  013701  033364              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8943  033346  012704  000125              MOV     #125,R4         ;RESULT S / B = 125
8944  033352  012705  064557              MOV     #DBTA+3,R5      ;SOURCE ADDR = DBTA+3
8945  033356  012703  177400              MOV     #177400,R3      ;[DEST] = 177400
8946  033362  000257                      CCC                     ;SCOPE SYNC
8947
8948  033364  111503              2$:     MOVB    (R5),R3         ;TEST THE MOVB
8949
8950  033366  020403                      CMP     R4,R3           ;RESULT CORRECT ?
8951  033370  001401                      BEQ     TST476          ;;BR IF YES
```

```
8952
8953  033372  104002                3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
8954
8955                                  ;;*****************************************************************
8956                                  ;*TEST 476       MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
8957                                  ;;*****************************************************************
8958  033374                          TST476:
8959  033374  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8960  033376  012700  000476                MOV     #476,R0         ;;LOAD R0 WITH TEST NUMBER
8961  033402  013701  033422                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8962  033406  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
8963  033412  012705  064555                MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
8964  033416  005003                        CLR     R3              ;[DEST] = 000000
8965  033420  000257                        CCC                     ;SCOPE SYNC
8966
8967  033422  112503                2$:     MOVB    (R5)+,R3        ;TEST THE MOVB
8968
8969  033424  020403                        CMP     R4,R3           ;RESULT CORRECT ?
8970  033426  001401                        BEQ     4$              ;BR IF YES
8971
8972  033430  104002                3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
8973
8974  033432  022705  064556        4$:     CMP     #DBTA+2,R5      ;DID MOVB INCREMENT SRC REG ?
8975  033436  001401                        BEQ     TST477          ;;BR IF YES
8976
8977  033440  104005                5$:     ERROR   5               ;MOVB FAILED TO UPDATE SRC REG
8978
8979                                  ;;*****************************************************************
8980                                  ;*TEST 477       MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
8981                                  ;;*****************************************************************
8982  033442                          TST477:
8983  033442  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8984  033444  012700  000477                MOV     #477,R0         ;;LOAD R0 WITH TEST NUMBER
8985  033450  013701  033470                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8986  033454  005004                        CLR     R4              ;RESULT S / B = 000000
8987  033456  012705  064554                MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
8988  033462  012703  177400                MOV     #177400,R3      ;[DEST] = 177400
8989  033466  000257                        CCC                     ;SCOPE SYNC
8990
8991  033470  112503                2$:     MOVB    (R5)+,R3                ;TEST THE MOVB
8992
8993  033472  020403                        CMP     R4,R3           ;RESULT CORRECT ?
8994  033474  001401                        BEQ     4$              ;BR IF YES
8995
8996  033476  104002                3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
8997
8998  033500  022705  064555        4$:     CMP     #DBTA+1,R5      ;DID MOVB INCREMENT SRC REG ?
8999  033504  001401                        BEQ     TST500          ;;BR IF YES
9000
9001  033506  104005                5$:     ERROR   5               ;MOVB FAILED TO UPDATE SOURCE REG
9002
9003                                  ;;*****************************************************************
9004                                  ;*TEST 500       MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
9005                                  ;;*****************************************************************
9006  033510                          TST500:
9007  033510  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
9008  033512  012700  000500          MOV     #500,R0         ;;LOAD R0 WITH TEST NUMBER
9009  033516  013701  033542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9010  033522  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9011  033526  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
9012  033532  012705  064555          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9013  033536  005012                  CLR     (R2)            ;[DEST] = 000000
9014  033540  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
9015
9016  033542  111512          2$:     MOVB    (R5),(R2)       ;TEST THE MOVB
9017
9018  033544  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
9019  033546  001402                  BEQ     TST501          ;;BR IF YES
9020
9021  033550  011203                  MOV     (R2),R3         ;GET THE WAS DATA
9022  033552  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
9023
9024                          ;;**************************************************************
9025                          ;*TEST 501      MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
9026                          ;;**************************************************************
9027  033554                  TST501:
9028  033554  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9029  033556  012700  000501          MOV     #501,R0         ;;LOAD R0 WITH TEST NUMBER
9030  033562  013701  033610          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9031  033566  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9032  033572  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
9033  033576  012705  064555          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9034  033602  005012                  CLR     (R2)            ;[DEST] = 000000
9035  033604  010203                  MOV     R2,R3           ;[R3] = DEST ADDR
9036  033606  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
9037
9038  033610  111523          2$:     MOVB    (R5),(R3)+      ;TEST THE MOVB
9039
9040  033612  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
9041  033614  001402                  BEQ     4$              ;BR IF YES
9042
9043  033616  011203                  MOV     (R2),R3         ;GET THE WAS DATA
9044  033620  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
9045
9046  033622  022703  063237  4$:     CMP     #MBUF0+1,R3     ;DID MOVB INCREMENT THE DEST REG ?
9047  033626  001401                  BEQ     TST502          ;;BR IF YES
9048
9049  033630  104005          5$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
9050
9051                          ;;**************************************************************
9052                          ;*TEST 502      MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
9053                          ;;**************************************************************
9054  033632                  TST502:
9055  033632  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9056  033634  012700  000502          MOV     #502,R0         ;;LOAD R0 WITH TEST NUMBER
9057  033640  013701  033670          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9058  033644  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9059  033650  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
9060  033654  012705  064555          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9061  033660  005012                  CLR     (R2)            ;[DEST] = 000000
9062  033662  012703  063232          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA +10
9063  033666  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
```

```
9064
9065  033670  111533                2$:     MOVB    (R5),@(R3)+     ;TEST THE MOVB
9066
9067  033672  022703  063234                CMP     #ATA+12,R3      ;DID DEST REG GET INCREMENTED ?
9068  033676  001401                        BEQ     4$              ;BR IF YES
9069
9070  033700  104005                3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
9071
9072  033702  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
9073  033704  001402                        BEQ     TST503          ;;BR IF YES
9074
9075  033706  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9076  033710  104001                5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
9077
9078                                ;;*****************************************************************
9079                                ;*TEST 503      MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
9080                                ;;*****************************************************************
9081  033712                        TST503:
9082  033712  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9083  033714  012700  000503                MOV     #503,R0         ;;LOAD R0 WITH TEST NUMBER
9084  033720  013701  033750                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9085  033724  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9086  033730  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
9087  033734  012705  064555                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9088  033740  005012                        CLR     (R2)            ;[DEST] = 000000
9089  033742  012703  063237                MOV     #MBUF0+1,R3     ;INITIAL DEST ADDR = MBUF0+1
9090  033746  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
9091
9092  033750  111543                2$:     MOVB    (R5),-(R3)      ;TEST THE MOVB
9093
9094  033752  020302                        CMP     R3,R2           ;DID MOVB DECREMENT DEST REG ?
9095  033754  001401                        BEQ     4$              ;BR IF YES
9096
9097  033756  104005                3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
9098
9099  033760  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
9100  033762  001402                        BEQ     TST504          ;;BR IF YES
9101
9102  033764  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9103  033766  104001                5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
9104
9105                                ;;*****************************************************************
9106                                ;*TEST 504      MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
9107                                ;;*****************************************************************
9108  033770                        TST504:
9109  033770  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9110  033772  012700  000504                MOV     #504,R0         ;;LOAD R0 WITH TEST NUMBER
9111  033776  013701  034026                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9112  034002  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9113  034006  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
9114  034012  012705  064555                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9115  034016  005012                        CLR     (R2)            ;[DEST] = 000000
9116  034020  012703  063234                MOV     #ATA+12,R3      ;INITIAL DEST ADDR = ATA +12
9117  034024  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
9118
9119  034026  111553                2$:     MOVB    (R5),@-(R3)     ;TEST THE MOVB
```

```
9120
9121  034030  022703  063232                CMP     #ATA+10,R3      ;DID MOVB DECREMENT DEST REG ?
9122  034034  001401                        BEQ     4$              ;BR IF YES
9123
9124  034036  104005              3$:       ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
9125
9126  034040  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
9127  034042  001402                        BEQ     TST505          ;;BR IF YES
9128
9129  034044  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9130  034046  104001              5$:       ERROR   1               ;MOVB DELIVERED WRONG RESULT
9131
9132                                        ;;*********************************************************************
9133                                        ;*TEST 505      MOVB TEST - SM1,DM6 - SRC ADR ODD / DST ADR EVEN
9134                                        ;*********************************************************************
9135  034050                                TST505:
9136  034050  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9137  034052  012700  000505                MOV     #505,R0         ;;LOAD R0 WITH TEST NUMBER
9138  034056  013701  034106                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9139  034062  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9140  034066  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
9141  034072  012705  064555                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9142  034076  005012                        CLR     (R2)            ;[DEST] = 000000
9143  034100  012703  063244                MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
9144  034104  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
9145
9146  034106  111563  177772    2$:         MOVB    (R5),-6(R3)     ;TEST THE MOVB
9147
9148  034112  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
9149  034114  001402                        BEQ     TST506          ;;BR IF YES
9150
9151  034116  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9152  034120  104001              3$:       ERROR   1               ;MOVB DELIVERED WRONG RESULT
9153
9154                                        ;;*********************************************************************
9155                                        ;*TEST 506      MOVB TEST - SM1,DM7 - SRC ADR ODD / DST ADR EVEN
9156                                        ;*********************************************************************
9157  034122                                TST506:
9158  034122  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9159  034124  012700  000506                MOV     #506,R0         ;;LOAD R0 WITH TEST NUMBER
9160  034130  013701  034160                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9161  034134  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
9162  034140  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
9163  034144  012705  064555                MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
9164  034150  005012                        CLR     (R2)            ;[DEST] = 000000
9165  034152  012703  063222                MOV     #ATA,R3         ;BASE DEST ADDR = ATA
9166  034156  000257                        CCC                     ;CLEAR FLAGS - SCOPE SYNC
9167
9168  034160  111573  000010    2$:         MOVB    (R5),@10(R3)    ;TEST THE MOVB
9169
9170  034164  020412                        CMP     R4,(R2)         ;CORRECT RESULT ?
9171  034166  001402                        BEQ     TST507          ;;BR IF YES
9172
9173  034170  011203                        MOV     (R2),R3         ;GET THE WAS DATA
9174  034172  104001              3$:       ERROR   1               ;MOVB DELIVERED WRONG RESULT
9175
```

```
 9176                                   ;;*****************************************************************
 9177                                   ;*TEST 507      MOVB  SM0,DM1 TEST
 9178                                   ;;*****************************************************************
 9179  034174                           TST507:
 9180  034174  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9181  034176  012700  000507                  MOV     #507,R0         ;;LOAD R0 WITH TEST NUMBER
 9182  034202  013701  034230                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9183  034206  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9184  034212  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 9185  034216  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9186  034222  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9187  034224  005012                          CLR     (R2)            ;[DEST] = 000000
 9188  034226  000257                          CCC                     ;SCOPE SYNC
 9189
 9190  034230  110315                  2$:     MOVB    R3,(R5)         ;TEST THE MOVB
 9191
 9192  034232  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9193  034234  001402                          BEQ     TST510          ;;BR IF YES
 9194
 9195  034236  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9196  034240  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9197
 9198                                   ;;*****************************************************************
 9199                                   ;*TEST 510      MOVB  SM0,DM2 TEST
 9200                                   ;;*****************************************************************
 9201  034242                           TST510:
 9202  034242  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9203  034244  012700  000510                  MOV     #510,R0         ;;LOAD R0 WITH TEST NUMBER
 9204  034250  013701  034276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9205  034254  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9206  034260  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 9207  034264  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9208  034270  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9209  034272  005012                          CLR     (R2)            ;[DEST] = 000000
 9210  034274  000257                          CCC                     ;SCOPE SYNC
 9211
 9212  034276  110325                  2$:     MOVB    R3,(R5)+        ;TEST THE MOVB
 9213
 9214  034300  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9215  034302  001402                          BEQ     TST511          ;;BR IF YES
 9216
 9217  034304  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9218  034306  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9219
 9220                                   ;;*****************************************************************
 9221                                   ;*TEST 511      MOVB  SM0,DM3 TEST
 9222                                   ;;*****************************************************************
 9223  034310                           TST511:
 9224  034310  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9225  034312  012700  000511                  MOV     #511,R0         ;;LOAD R0 WITH TEST NUMBER
 9226  034316  013701  034346                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9227  034322  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9228  034326  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 9229  034332  012703  177777                  MOV     #-1,R3          ;SOURCE OP IN R3
 9230  034336  012705  063232                  MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9231  034342  005012                          CLR     (R2)            ;[DEST] = 000000
```

```
 9232  034344  000257                          CCC                     ;SCOPE SYNC
 9233
 9234  034346  110335                  2$:     MOVB    R3,@(R5)+       ;TEST THE MOVB
 9235
 9236  034350  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9237  034352  001402                          BEQ     TST512          ;;BR IF YES
 9238
 9239  034354  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9240  034356  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9241
 9242                                  ;;**************************************************************
 9243                                  ;*TEST 512       MOVB  SM0,DM4 TEST
 9244                                  ;;**************************************************************
 9245  034360                          TST512:
 9246  034360  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9247  034362  012700  000512                  MOV     #512,R0         ;;LOAD R0 WITH TEST NUMBER
 9248  034366  013701  034416                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9249  034372  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9250  034376  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
 9251  034402  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9252  034406  012705  063240                  MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9253  034412  005012                          CLR     (R2)            ;[DEST] = 000000
 9254  034414  000257                          CCC                     ;SCOPE SYNC
 9255
 9256  034416  110345                  2$:     MOVB    R3,-(R5)        ;TEST THE MOVB
 9257
 9258  034420  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9259  034422  001402                          BEQ     TST513          ;;BR IF YES
 9260
 9261  034424  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9262  034426  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9263
 9264                                  ;;**************************************************************
 9265                                  ;*TEST 513       MOVB  SM0,DM6 TEST
 9266                                  ;;**************************************************************
 9267  034430                          TST513:
 9268  034430  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9269  034432  012700  000513                  MOV     #513,R0         ;;LOAD R0 WITH TEST NUMBER
 9270  034436  013701  034466                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9271  034442  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9272  034446  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 9273  034452  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9274  034456  012705  063240                  MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9275  034462  005012                          CLR     (R2)            ;[DEST] = 000000
 9276  034464  000257                          CCC                     ;SCOPE SYNC
 9277
 9278  034466  110365  177776          2$:     MOVB    R3,-2(R5)       ;TEST THE MOVB
 9279
 9280  034472  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9281  034474  001402                          BEQ     TST514          ;;BR IF YES
 9282
 9283  034476  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9284  034500  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9285
 9286                                  ;;**************************************************************
 9287                                  ;*TEST 514       BIS TEST - SM0,DM0 - N:C = 0111
```

```
9288                                    ;;*****************************************************************
9289  034502                            TST514:
9290  034502  000004                            SCOPE                           ;CALL THE SCOPE LOOP UTILITY
9291  034504  012700  000514                    MOV     #514,R0                 ;;LOAD R0 WITH TEST NUMBER
9292  034510  013701  034534                    MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
9293  034514  012704  177777                    MOV     #-1,R4                  ;RESULT S / B = 177777
9294  034520  012705  125252                    MOV     #125252,R5              ;SRC OPR = 125252
9295  034524  012703  052525                    MOV     #52525,R3               ;[DEST] = 52525
9296  034530  000257                            CCC                             ;CLEAR FLAGS
9297  034532  000267                            267                             ;N:C = 0111
9298
9299  034534  050503                    2$:     BIS     R5,R3                   ;TEST THE BIS
9300
9301  034536  100003                            BPL     3$                      ;N:C = 1001 ?
9302  034540  001402                            BEQ     3$
9303  034542  102401                            BVS     3$
9304  034544  103401                            BCS     4$
9305
9306  034546  104002                    3$:     ERROR   2                       ;BIS FAILED TO ALTER CODES PROPERLY
9307
9308  034550  020403                    4$:     CMP     R4,R3                   ;CORRECT RESULT ?
9309  034552  001401                            BEQ     TST515                  ;;BR IF YES
9310
9311  034554  104002                    5$:     ERROR   2                       ;BIS DELIVERED THE WRONG RESULT
9312
9313                                    ;;*****************************************************************
9314                                    ;*TEST 515      BIS TEST - SM0,DM0 - N:C = 1000
9315                                    ;;*****************************************************************
9316  034556                            TST515:
9317  034556  000004                            SCOPE                           ;CALL THE SCOPE LOOP UTILITY
9318  034560  012700  000515                    MOV     #515,R0                 ;;LOAD R0 WITH TEST NUMBER
9319  034564  013701  034614                    MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
9320                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT10
9321  034570  032737  002000  063160            BIT     #BIT10,@#BPTLOC         ;BREAKPOINT HALT SET ??
9322  034576  001401                            BEQ     .+4                     ;BR IF NOT
9323  034600  000000                            HALT                            ;BREAK-DEPRESS CONTINUE TO CONTINUE
9324  034602  005004                            CLR     R4                      ;RESULT S / B = 000000
9325  034604  005005                            CLR     R5                      ;SRC OPR = 000000
9326  034606  005003                            CLR     R3                      ;[DEST] = 000000
9327  034610  000257                            CCC                             ;CLEAR FLAGS
9328  034612  000270                            SEN                             ;N:C = 1000
9329
9330  034614  050503                    2$:     BIS     R5,R3                   ;TEST THE BIS
9331
9332  034616  100403                            BMI     3$                      ;N:C = 0100
9333  034620  001002                            BNE     3$
9334  034622  102401                            BVS     3$
9335  034624  103001                            BCC     4$
9336
9337  034626  104002                    3$:     ERROR   2                       ;BIS FAILED TO ALTER CODES PROPERLY
9338
9339  034630  020403                    4$:     CMP     R4,R3                   ;CORRECT RESULT ?
9340  034632  001401                            BEQ     TST516                  ;;BR IF YES
9341
9342  034634  104002                    5$:     ERROR   2                       ;BIS DELIVERED THE WRONG RESULT
9343
```

```
9344                                    ;;*****************************************************************
9345                                    ;*TEST 516      BIC TEST - SM0,DM0 - N:C = 0111
9346                                    ;;*****************************************************************
9347  034636                            TST516:
9348  034636  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9349  034640  012700  000516                    MOV     #516,R0         ;;LOAD R0 WITH TEST NUMBER
9350  034644  013701  034670                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9351  034650  012704  100000                    MOV     #100000,R4      ;RESULT S / B = 100000
9352  034654  012705  077777                    MOV     #77777,R5       ;SRC OPR = 77777
9353  034660  012703  177777                    MOV     #-1,R3          ;[DEST] = 177777
9354  034664  000257                            CCC                     ;CLEAR FLAGS
9355  034666  000267                            267                     ;N:C = 0111
9356
9357  034670  040503                    2$:     BIC     R5,R3           ;TEST THE BIC
9358
9359  034672  100003                            BPL     3$              ;N:C = 1001 ?
9360  034674  001402                            BEQ     3$
9361  034676  102401                            BVS     3$
9362  034700  103401                            BCS     4$
9363
9364  034702  104002                    3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
9365
9366  034704  020403                    4$:     CMP     R4,R3           ;CORRECT RESULT ?
9367  034706  001401                            BEQ     TST517          ;;BR IF YES
9368
9369  034710  104002                    5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
9370
9371                                    ;;*****************************************************************
9372                                    ;*TEST 517      BIC TEST - SM0,DM0 - N:C = 1000
9373                                    ;;*****************************************************************
9374  034712                            TST517:
9375  034712  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9376  034714  012700  000517                    MOV     #517,R0         ;;LOAD R0 WITH TEST NUMBER
9377  034720  013701  034736                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9378  034724  005004                            CLR     R4              ;RESULT S / B = 000000
9379  034726  005005                            CLR     R5              ;SRC OPR = 000000
9380  034730  005003                            CLR     R3              ;[DEST] = 000000
9381  034732  000257                            CCC                     ;CLEAR FLAGS
9382  034734  000270                            SEN                     ;N:C = 1000
9383
9384  034736  040503                    2$:     BIC     R5,R3           ;TEST THE BIC
9385
9386  034740  100403                            BMI     3$              ;N:C = 0100
9387  034742  001002                            BNE     3$
9388  034744  102401                            BVS     3$
9389  034746  103001                            BCC     4$
9390
9391  034750  104002                    3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
9392
9393  034752  020403                    4$:     CMP     R4,R3           ;CORRECT RESULT ?
9394  034754  001401                            BEQ     TST520          ;;BR IF YES
9395
9396  034756  104002                    5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
9397
9398                                    ;;*****************************************************************
9399                                    ;*TEST 520      BIT TEST - SM0,DM0 - N:C = 0111
```

```
 9400                                   ;;*****************************************************************
 9401  034760                           TST520:
 9402  034760  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9403  034762  012700  000520                   MOV     #520,R0         ;;LOAD R0 WITH TEST NUMBER
 9404  034766  013701  035012                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9405  034772  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
 9406  034776  012705  100000                   MOV     #100000,R5      ;SRC OPR = 100000
 9407  035002  012703  100000                   MOV     #100000,R3      ;[DEST] = 100000
 9408  035006  000257                           CCC                     ;CLEAR FLAGS
 9409  035010  000267                           267                     ;N:C = 0111
 9410
 9411  035012  030503                   2$:     BIT     R5,R3           ;TEST THE BIT
 9412
 9413  035014  100003                           BPL     3$              ;N:C = 1001
 9414  035016  001402                           BEQ     3$
 9415  035020  102401                           BVS     3$
 9416  035022  103401                           BCS     4$
 9417
 9418  035024  104002                   3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
 9419
 9420  035026  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9421  035030  001402                           BEQ     TST521          ;;BR IF YES
 9422
 9423  035032  011203                           MOV     (R2),R3         ;GET THE WAS DATA
 9424  035034  104002                   5$:     ERROR   2               ;BIT DELIVERED A RESULT
 9425
 9426                                   ;;*****************************************************************
 9427                                   ;TEST 521       BIT TEST - SMO,DMO - N:C = 1000
 9428                                   ;;*****************************************************************
 9429  035036                           TST521:
 9430  035036  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9431  035040  012700  000521                   MOV     #521,R0         ;;LOAD R0 WITH TEST NUMBER
 9432  035044  013701  035066                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9433  035050  012704  125252                   MOV     #125252,R4      ;RESULT S / B = 125252
 9434  035054  012705  052525                   MOV     #52525,R5       ;SRC OPR = 52525
 9435  035060  010403                           MOV     R4,R3           ;[DEST] = 125252
 9436  035062  000257                           CCC                     ;CLEAR FLAGS
 9437  035064  000270                           SEN                     ;N:C = 1000
 9438
 9439  035066  030503                   2$:     BIT     R5,R3   ;TEST THE BIT
 9440
 9441  035070  100403                           BMI     3$              ;N:C = 0100
 9442  035072  001002                           BNE     3$
 9443  035074  102401                           BVS     3$
 9444  035076  103001                           BCC     4$
 9445
 9446  035100  104002                   3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
 9447
 9448  035102  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9449  035104  001401                           BEQ     TST522          ;;BR IF YES
 9450
 9451  035106  104002                   5$:     ERROR   2               ;BIT DELIVERED A RESULT
 9452
 9453                                   ;;*****************************************************************
 9454                                   ;TEST 522       CMP TEST - SMO,DMO - N:C = 0110
 9455                                   ;;*****************************************************************
```

```
9456  035110                          TST522:
9457  035110  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9458  035112  012700  000522                  MOV     #522,R0         ;;LOAD R0 WITH TEST NUMBER
9459  035116  013701  035140                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9460  035122  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
9461  035126  005005                          CLR     R5              ;SRC OPR = 000000
9462  035130  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
9463  035134  000257                          CCC                     ;CLEAR FLAGS
9464  035136  000266                          266                     ;N:C = 0110
9465
9466  035140  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
9467
9468  035142  100003                          BPL     3$              ;N:C = 1001
9469  035144  001402                          BEQ     3$
9470  035146  102401                          BVS     3$
9471  035150  103401                          BCS     4$
9472
9473  035152  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
9474
9475  035154  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
9476  035156  001401                          BEQ     TST523          ;;BR IF YES
9477
9478  035160  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
9479
9480                                  ;;**********************************************************************
9481                                  ;*TEST 523       CMP TEST - SM0,DM0 - N:C = 1010
9482                                  ;;**********************************************************************
9483  035162                          TST523:
9484  035162  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9485  035164  012700  000523                  MOV     #523,R0         ;;LOAD R0 WITH TEST NUMBER
9486  035170  013701  035212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9487  035174  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
9488  035200  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
9489  035204  010403                          MOV     R4,R3           ;[DEST] = 177777
9490  035206  000257                          CCC                     ;CLEAR FLAGS
9491  035210  000272                          272                     ;N:C = 1010
9492
9493  035212  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
9494
9495  035214  100403                          BMI     3$              ;N:C = 0100
9496  035216  001002                          BNE     3$
9497  035220  102401                          BVS     3$
9498  035222  103001                          BCC     4$
9499
9500  035224  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
9501
9502  035226  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
9503  035230  001401                          BEQ     TST524          ;;BR IF YES
9504
9505  035232  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
9506
9507                                  ;;**********************************************************************
9508                                  ;*TEST 524       CMP TEST - SM0,DM0 - N:C = 0000
9509                                  ;;**********************************************************************
9510  035234                          TST524:
9511  035234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9512  035236  012700  000524                MOV     #524,R0         ;;LOAD R0 WITH TEST NUMBER
 9513  035242  013701  035264                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9514  035246  012704  000001                MOV     #+1,R4          ;RESULT S / B = +1
 9515  035252  012705  100000                MOV     #100000,R5                      ;SRC OPR = 100000
 9516  035256  012703  000001                MOV     #+1,R3          ;[DEST] = +1
 9517  035262  000257                        CCC                     ;CLEAR FLAGS
 9518
 9519  035264  020503                2$:     CMP     R5,R3           ;TEST THE CMP
 9520
 9521  035266  100403                        BMI     3$              ;N:C = 0010
 9522  035270  001402                        BEQ     3$
 9523  035272  102001                        BVC     3$
 9524  035274  103001                        BCC     4$
 9525
 9526  035276  104002                3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
 9527
 9528  035300  020403                4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9529  035302  001401                        BEQ     TST525          ;;BR IF YES
 9530
 9531  035304  104002                5$:     ERROR   2               ;CMP DELIVERED A RESULT
 9532
 9533                                ;;**************************************************************
 9534                                ;*TEST 525      BIS TEST - SM0,DM1 - N:C = 0111
 9535                                ;;**************************************************************
 9536  035306                        TST525:
 9537  035306  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9538  035310  012700  000525                MOV     #525,R0         ;;LOAD R0 WITH TEST NUMBER
 9539  035314  013701  035344                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9540  035320  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9541  035324  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
 9542  035330  012705  125252                MOV     #125252,R5      ;SRC OPR = 125252
 9543  035334  012712  052525                MOV     #52525,(R2)     ;[DEST] = 52525
 9544  035340  000257                        CCC                     ;CLEAR FLAGS
 9545  035342  000267                        267                     ;N:C = 0111
 9546
 9547  035344  050512                2$:     BIS     R5,(R2)         ;TEST THE BIS
 9548
 9549  035346  100003                        BPL     3$              ;N:C = 1001
 9550  035350  001402                        BEQ     3$
 9551  035352  102401                        BVS     3$
 9552  035354  103401                        BCS     4$
 9553
 9554  035356  104001                3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
 9555
 9556  035360  020412                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9557  035362  001402                        BEQ     TST526          ;;BR IF YES
 9558
 9559  035364  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9560  035366  104001                5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
 9561
 9562                                ;;**************************************************************
 9563                                ;*TEST 526      BIS TEST - SM0,DM1 - N:C = 1000
 9564                                ;;**************************************************************
 9565  035370                        TST526:
 9566  035370  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9567  035372  012700  000526                MOV     #526,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 9568  035376  013701  035420                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9569  035402  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9570  035406  005004                        CLR     R4              ;RESULT S / B = 000000
 9571  035410  005005                        CLR     R5              ;SRC OPR = 000000
 9572  035412  005012                        CLR     (R2)            ;[DEST] = 000000
 9573  035414  000257                        CCC                     ;CLEAR FLAGS
 9574  035416  000270                        SEN                     ;N:C = 1000
 9575
 9576  035420  050512              2$:       BIS     R5,(R2)         ;TEST THE BIS
 9577
 9578  035422  100403                        BMI     3$              ;N:C = 0100
 9579  035424  001002                        BNE     3$
 9580  035426  102401                        BVS     3$
 9581  035430  103001                        BCC     4$
 9582
 9583  035432  104001              3$:       ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
 9584
 9585  035434  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 9586  035436  001402                        BEQ     TST527          ;;BR IF YES
 9587
 9588  035440  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9589  035442  104001              5$:       ERROR   1               ;BIS DELIVERED THE WRONG RESULT
 9590
 9591                              ;;**************************************************************
 9592                              ;*TEST 527      BIC TEST - SM0,DM1 - N:C = 0111
 9593                              ;;**************************************************************
 9594  035444                    TST527:
 9595  035444  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9596  035446  012700  000527                MOV     #527,R0         ;;LOAD R0 WITH TEST NUMBER
 9597  035452  013701  035502                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9598  035456  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9599  035462  012704  100000                MOV     #100000,R4      ;RESULT S / B = 100000
 9600  035466  012705  077777                MOV     #77777,R5               ;SRC OPR = 77777
 9601  035472  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
 9602  035476  000257                        CCC                     ;CLEAR FLAGS
 9603  035500  000267                        267                     ;N:C = 0111
 9604
 9605  035502  040512              2$:       BIC     R5,(R2)         ;TEST THE BIC
 9606
 9607  035504  100003                        BPL     3$              ;N:C = 1001
 9608  035506  001402                        BEQ     3$
 9609  035510  102401                        BVS     3$
 9610  035512  103401                        BCS     4$
 9611
 9612  035514  104001              3$:       ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
 9613
 9614  035516  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 9615  035520  001402                        BEQ     TST530          ;;BR IF YES
 9616
 9617  035522  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9618  035524  104001              5$:       ERROR   1               ;BIC DELIVERED THE WRONG RESULT
 9619
 9620                              ;;**************************************************************
 9621                              ;*TEST 530      BIC TEST - SM0,DM1 - N:C = 1000
 9622                              ;;**************************************************************
 9623  035526                    TST530:
```

```
 9624  035526  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9625  035530  012700  000530                MOV     #530,R0         ;;LOAD R0 WITH TEST NUMBER
 9626  035534  013701  035556                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9627  035540  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9628  035544  005004                        CLR     R4              ;RESULT S / B = 000000
 9629  035546  005005                        CLR     R5              ;SRC OPR = 000000
 9630  035550  005012                        CLR     (R2)            ;[DEST] = 000000
 9631  035552  000257                        CCC                     ;CLEAR FLAGS
 9632  035554  000270                        SEN                     ;N:C = 1000
 9633
 9634  035556  040512              2$:       BIC     R5,(R2)         ;TEST THE BIC
 9635
 9636  035560  100403                        BMI     3$              ;N:C = 0100
 9637  035562  001002                        BNE     3$
 9638  035564  102401                        BVS     3$
 9639  035566  103001                        BCC     4$
 9640
 9641  035570  104001              3$:       ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
 9642
 9643  035572  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 9644  035574  001402                        BEQ     TST531          ;;BR IF YES
 9645
 9646  035576  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9647  035600  104001              5$:       ERROR   1               ;BIC DELIVERED THE WRONG RESULT
 9648
 9649                              ;;**************************************************************
 9650                              ;*TEST 531      BIT TEST - SM0,DM1 - N:C = 0111
 9651                              ;;**************************************************************
 9652  035602                      TST531:
 9653  035602  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9654  035604  012700  000531                MOV     #531,R0         ;;LOAD R0 WITH TEST NUMBER
 9655  035610  013701  035640                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9656  035614  012702  063236                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9657  035620  012704  100000                MOV     #100000,R4      ;RESULT S / B = 100000
 9658  035624  012705  100000                MOV     #100000,R5      ;SRC OPR = 100000
 9659  035630  012712  100000                MOV     #100000,(R2)    ;[DEST] = 100000
 9660  035634  000257                        CCC                     ;CLEAR FLAGS
 9661  035636  000267                        267                     ;N:C = 0111
 9662
 9663  035640  030512              2$:       BIT     R5,(R2)         ;TEST THE BIT
 9664
 9665  035642  100003                        BPL     3$              ;N:C = 1001
 9666  035644  001402                        BEQ     3$
 9667  035646  102401                        BVS     3$
 9668  035650  103401                        BCS     4$
 9669
 9670  035652  104001              3$:       ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
 9671
 9672  035654  020412              4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
 9673  035656  001402                        BEQ     TST532          ;;BR IF YES
 9674
 9675  035660  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 9676  035662  104001              5$:       ERROR   1               ;BIT DELIVERED A RESULT
 9677
 9678                              ;;**************************************************************
 9679                              ;*TEST 532      BIT TEST - SM0,DM1 - N:C = 1000
```

```
                                ;;*****************************************************************
035664                          TST532:
035664  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
035666  012700  000532                 MOV     #532,R0         ;;LOAD R0 WITH TEST NUMBER
035672  013701  035722                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
035676  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
035702  012704  052525                 MOV     #52525,R4       ;RESULT S / B = 52525
035706  012705  125252                 MOV     #125252,R5      ;SRC OPR = 125252
035712  012712  052525                 MOV     #52525,(R2)     ;[DEST] = 52525
035716  000257                          CCC                     ;CLEAR FLAGS
035720  000270                          SEN                     ;N:C = 1000
                                
035722  030512                  2$.     BIT     R5,(R2)         ;TEST THE BIT
                                
035724  100403                          BMI     3$              ;N:C = 0100
035726  001002                          BNE     3$
035730  102401                          BVS     3$
035732  103001                          BCC     4$
                                
035734  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
                                
035736  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
035740  001402                          BEQ     TST533          ;;BR IF YES
                                
035742  011203                          MOV     (R2),R3         ;GET THE WAS DATA
035744  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
                                ;;*****************************************************************
                                ;TEST 533       CMP TEST - SM0,DM1 - N:C = 1010
                                ;;*****************************************************************
035746                          TST533:
035746  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
035750  012700  000533                 MOV     #533,R0         ;;LOAD R0 WITH TEST NUMBER
035754  013701  036004                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
035760  012702  063236                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
035764  012704  177777                 MOV     #-1,R4          ;RESULT S / B = -1
035770  012705  177777                 MOV     #-1,R5          ;SRC OPR = 177777
035774  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
036000  000257                          CCC                     ;CLEAR FLAGS
036002  000272                          272                     ;N:C = 1010
                                
036004  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
                                
036006  100403                          BMI     3$              ;N:C = 0100
036010  001002                          BNE     3$
036012  102401                          BVS     3$
036014  103001                          BCC     4$
                                
036016  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
                                
036020  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
036022  001402                          BEQ     TST534          ;;BR IF YES
                                
036024  011203                          MOV     (R2),R3         ;GET THE WAS DATA
036026  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
                                
                                ;;*****************************************************************
```

```
                                    ;*TEST 534      CMP TEST - SM0,DM1 - N:C = 0110
                                    ;*****************************************************************
036030                              TST534:
036030  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
036032  012700  000534                      MOV     #534,R0         ;;LOAD R0 WITH TEST NUMBER
036036  013701  036064                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
036042  012702  063236                      MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
036046  012704  000001                      MOV     #+1,R4          ;RESULT S / B = +1
036052  005005                              CLR     R5              ;SRC OPR = 000000
036054  012712  000001                      MOV     #+1,(R2)        ;[DEST] = +1
036060  000257                              CCC                     ;CLEAR FLAGS
036062  000266                              266                     ;N:C = 0110

036064  020512                    2$:       CMP     R5,(R2)         ;TEST THE CMP

036066  100003                              BPL     3$              ;N:C = 1001
036070  001402                              BEQ     3$
036072  102401                              BVS     3$
036074  103401                              BCS     4$

036076  104001                    3$:       ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY

036100  020412                    4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
036102  001402                              BEQ     TST535          ;;BR IF YES

036104  011203                              MOV     (R2),R3         ;GET THE WAS DATA
036106  104001                    5$:       ERROR   1               ;CMP DELIVERED A RESULT

                                    ;*****************************************************************
                                    ;*TEST 535      CMP TEST - SM0,DM1 - N:C = 0000
                                    ;*****************************************************************
036110                              TST535:
036110  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
036112  012700  000535                      MOV     #535,R0         ;;LOAD R0 WITH TEST NUMBER
036116  013701  036144                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
036122  012702  063236                      MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
036126  012704  000001                      MOV     #+1,R4          ;RESULT S / B = +1
036132  012705  100000                      MOV     #100000,R5      ;SRC OPR = 100000
036136  012712  000001                      MOV     #+1,(R2)        ;[DEST] = +1
036142  000257                              CCC                     ;CLEAR FLAGS

036144  020512                    2$:       CMP     R5,(R2)         ;TEST THE CMP

036146  100403                              BMI     3$              ;N:C = 0010
036150  001402                              BEQ     3$
036152  102001                              BVC     3$
036154  103001                              BCC     4$

036156  104001                    3$:       ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY

036160  020412                    4$:       CMP     R4,(R2)         ;CORRECT RESULT ?
036162  001402                              BEQ     TST536          ;;BR IF YES

036164  011203                              MOV     (R2),R3         ;GET THE WAS DATA
036166  104001                    5$:       ERROR   1               ;CMP DELIVERED A RESULT
```

```
9792                                    ;;*****************************************************************
9793                                    ;*TEST 536      BIS TEST - SM1,DM0 - N:C = 0111
9794                                    ;;*****************************************************************
9795  036170                            TST536:
9796  036170  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9797  036172  012700  000536                   MOV     #536,R0         ;;LOAD R0 WITH TEST NUMBER
9798  036176  013701  036222                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9799  036202  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
9800  036206  012705  063256                   MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
9801  036212  012703  052525                   MOV     #52525,R3       ;[DEST] = 52525
9802  036216  000257                           CCC                     ;CLEAR FLAGS
9803  036220  000267                           SCA                     ;N:C = 0111
9804
9805  036222  051503                    2$:    BIS     (R5),R3         ;TEST THE BIS
9806
9807  036224  100003                           BPL     3$              ;N:C = 1001
9808  036226  001402                           BEQ     3$
9809  036230  102401                           BVS     3$
9810  036232  103401                           BCS     4$
9811
9812  036234  104002                    3$:    ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
9813
9814  036236  020403                    4$:    CMP     R4,R3           ;CORRECT RESULT ?
9815  036240  001401                           BEQ     TST537          ;;BR IF YES
9816
9817  036242  104002                    5$:    ERROR   2               ;BIS DELIVERED THE WRONG RESULT
9818
9819                                    ;;*****************************************************************
9820                                    ;*TEST 537      BIS TEST - SM1,DM0 - N:C = 1000
9821                                    ;;*****************************************************************
9822  036244                            TST537:
9823  036244  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
9824  036246  012700  000537                   MOV     #537,R0         ;;LOAD R0 WITH TEST NUMBER
9825  036252  013701  036272                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
9826  036256  005004                           CLR     R4              ;RESULT S / B =000000
9827  036260  012705  063246                   MOV     #DWTA,R5        ;SRC ADDR = DWTA
9828  036264  005003                           CLR     R3              ;[DEST] = 000000
9829  036266  000257                           CCC                     ;CLEAR FLAGS
9830  036270  000270                           SEN                     ;N:C = 1000
9831
9832  036272  051503                    2$:    BIS     (R5),R3         ;TEST THE BIS
9833
9834  036274  100403                           BMI     3$              ;N:C = 0100
9835  036276  001002                           BNE     3$
9836  036300  102401                           BVS     3$
9837  036302  103001                           BCC     4$
9838
9839  036304  104002                    3$:    ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
9840
9841  036306  020403                    4$:    CMP     R4,R3           ;CORRECT RESULT ?
9842  036310  001401                           BEQ     TST540          ;;BR IF YES
9843
9844  036312  104002                    5$:    ERROR   2               ;BIS DELIVERED THE WRONG RESULT
9845
9846                                    ;;*****************************************************************
9847                                    ;*TEST 540      BIC TEST - SM1,DM0 - N:C = 0111
```

```
 9848                                   ;;*****************************************************************
 9849  036314                           TST540:
 9850  036314  000004                           SCOPE                           ;CALL THE SCOPE LOOP UTILITY
 9851  036316  012700  000540                   MOV     #540,R0                 ;;LOAD R0 WITH TEST NUMBER
 9852  036322  013701  036352                   MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
 9853  036326  012704  100000                   MOV     #100000,R4              ;RESULT S / B = 100000
 9854  036332  012705  063242                   MOV     #MBUF1,R5               ;SRC ADDR = MBUF1
 9855  036336  012703  177777                   MOV     #-1,R3  ;[DEST] = 177777
 9856  036342  012715  077777                   MOV     #77777,(R5)             ;SRC OPR = 77777
 9857  036346  000257                           CCC                             ;CLEAR FLAGS
 9858  036350  000267                           267                             ;N:C = 0111
 9859
 9860  036352  041503                 2$:       BIC     (R5),R3                 ;TEST THE BIC
 9861
 9862  036354  100003                           BPL     3$                      ;N:C = 1001 ?
 9863  036356  001402                           BEQ     3$
 9864  036360  102401                           BVS     3$
 9865  036362  103401                           BCS     4$
 9866
 9867  036364  104002                 3$:       ERROR   2                       ;BIC FAILED TO ALTER CODES PROPERLY
 9868
 9869  036366  020403                 4$:       CMP     R4,R3                   ;CORRECT RESULT ?
 9870  036370  001401                           BEQ     TST541                  ;;BR IF YES
 9871
 9872  036372  104002                 5$:       ERROR   2                       ;BIC DELIVERED THE WRONG RESULT
 9873
 9874                                   ;;*****************************************************************
 9875                                   ;;TEST 541      BIC TEST - SM1,DM0 - N:C = 1000
 9876                                   ;;*****************************************************************
 9877  036374                           TST541:
 9878  036374  000004                           SCOPE                           ;CALL THE SCOPE LOOP UTILITY
 9879  036376  012700  000541                   MOV     #541,R0                 ;;LOAD R0 WITH TEST NUMBER
 9880  036402  013701  036422                   MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WOR)
 9881  036406  005004                           CLR     R4                      ;RESULT S / B = 000000
 9882  036410  012705  063246                   MOV     #DWTA,R5                ;SRC ADDR = DWTA
 9883  036414  005003                           CLR     R3                      ;[DEST] =000000
 9884  036416  000257                           CCC                             ;CLEAR FLAGS
 9885  036420  000270                           SEN                             ;N:C = 1000
 9886
 9887  036422  041503                 2$:       BIC     (R5),R3                 ;TEST THE BIC
 9888
 9889  036424  100403                           BMI     3$                      ;N:C = 0100
 9890  036426  001002                           BNE     3$
 9891  036430  102401                           BVS     3$
 9892  036432  103001                           BCC     4$
 9893
 9894  036434  104002                 3$:       ERROR   2                       ;BIC FAILED TO ALTER CODES PROPERLY
 9895
 9896  036436  020403                 4$:       CMP     R4,R3                   ;CORRECT RESULT ?
 9897  036440  001401                           BEQ     TST542                  ;;BR IF YES
 9898
 9899  036442  104002                 5$:       ERROR   2                       ;BIC DELIVERED THE WRONG RESULT
 9900
 9901                                   ;;*****************************************************************
 9902                                   ;;TEST 542      BIT TEST - SM1,DM0 - N:C = 0111
 9903                                   ;;*****************************************************************
```

```
 9904  036444                          TST542:
 9905  036444  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9906  036446  012700  000542                  MOV     #542,R0         ;;LOAD R0 WITH TEST NUMBER
 9907  036452  013701  036474                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9908  036456  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
 9909  036462  012705  063250                  MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
 9910  036466  010403                          MOV     R4,R3           ;[DEST] = 100000
 9911  036470  000257                          CCC                     ;CLEAR FLAGS
 9912  036472  000267                          267                     ;N:C = 0111
 9913
 9914  036474  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
 9915
 9916  036476  100003                          BPL     3$              ;N:C = 1001 ?
 9917  036500  001402                          BEQ     3$
 9918  036502  102401                          BVS     3$
 9919  036504  103401                          BCS     4$
 9920
 9921  036506  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
 9922
 9923  036510  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9924  036512  001401                          BEQ     TST543          ;;BR IF YES
 9925
 9926  036514  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
 9927
 9928                                  ;;**************************************************************
 9929                                  ;*TEST 543      BIT TEST - SM1,DM0 - N:C = 1000
 9930                                  ;;**************************************************************
 9931  036516                          TST543:
 9932  036516  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9933  036520  012700  000543                  MOV     #543,R0         ;;LOAD R0 WITH TEST NUMBER
 9934  036524  013701  036546                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9935  036530  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
 9936  036534  012705  063256                  MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
 9937  036540  010403                          MOV     R4,R3           ;[DEST] = 52525
 9938  036542  000257                          CCC                     ;CLEAR FLAGS
 9939  036544  000270                          SEN                     ;N:C = 1000
 9940
 9941  036546  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
 9942
 9943  036550  100403                          BMI     3$              ;N:C = 0100
 9944  036552  001002                          BNE     3$
 9945  036554  102401                          BVS     3$
 9946  036556  103001                          BCC     4$
 9947
 9948  036560  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
 9949
 9950  036562  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9951  036564  001401                          BEQ     TST544          ;;BR IF YES
 9952
 9953  036566  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
 9954                                  ;;**************************************************************
 9955                                  ;*TEST 544      CMP TEST - SM1,DM0 - N:C = 0110
 9956                                  ;;**************************************************************
 9957  036570                          TST544:
 9958  036570  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9959  036572  012700  000544                  MOV     #544,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 9960  036576  013701  036620          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9961  036602  012704  000001          MOV     #+1,R4          ;RESULT S / B = +1
 9962  036606  012705  063246          MOV     #DWTA,R5        ;SRC ADDR = DWTA
 9963  036612  010403                  MOV     R4,R3           ;[DEST] = +1
 9964  036614  000257                  CCC                     ;CLEAR FLAGS
 9965  036616  000266                  266                     ;N:C = 0110
 9966
 9967  036620  021503          2$:     CMP     (R5),R3         ;TEST THE CMP
 9968
 9969  036622  100003                  BPL     3$              ;N:C = 1001
 9970  036624  001402                  BEQ     3$
 9971  036626  102401                  BVS     3$
 9972  036630  103401                  BCS     4$
 9973
 9974  036632  104002          3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
 9975
 9976  036634  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
 9977  036636  001401                  BEQ     TST545          ;;BR IF YES
 9978
 9979  036640  104002          5$:     ERROR   2               ;CMP DELIVERED A RESULT
 9980
 9981                          ;;**************************************************************
 9982                          ;*TEST 545      CMP TEST - SM1,DM0 - N:C = 1010
 9983                          ;;**************************************************************
 9984  036642                  TST545:
 9985  036642  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9986  036644  012700  000545          MOV     #545,R0         ;;LOAD R0 WITH TEST NUMBER
 9987  036650  013701  036672          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9988  036654  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 9989  036660  012705  063250          MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
 9990  036664  010403                  MOV     R4,R3           ;[DEST] = 177777
 9991  036666  000257                  CCC                     ;CLEAR FLAGS
 9992  036670  000272                  272                     ;N:C = 1010
 9993
 9994  036672  021503          2$:     CMP     (R5),R3         ;TEST THE CMP
 9995
 9996  036674  100403                  BMI     3$              ;N:C = 0100
 9997  036676  001002                  BNE     3$
 9998  036700  102401                  BVS     3$
 9999  036702  103001                  BCC     4$
10000
10001  036704  104002          3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10002
10003  036706  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10004  036710  001401                  BEQ     TST546          ;;BR IF YES
10005
10006  036712  104002          5$:     ERROR   2               ;CMP DELIVERED A RESULT
10007
10008                          ;;**************************************************************
10009                          ;*TEST 546      CMP TEST - SM1,DM0 - N:C = 0000
10010                          ;;**************************************************************
10011  036714                  TST546:
10012  036714  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10013  036716  012700  000546          MOV     #546,R0         ;;LOAD R0 WITH TEST NUMBER
10014  036722  013701  036750          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10015  036726  012704  000001          MOV     #+1,R4          ;RESULT S / B = +1
```

```
10016  036732  012705  063242                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10017  036736  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10018  036742  012715  100000                  MOV     #100000,(R5)    ;SRC OPR = 100000
10019  036746  000257                          CCC                     ;CLEAR FLAGS
10020
10021  036750  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10022
10023  036752  100403                          BMI     3$              ;N:C = 0010
10024  036754  001402                          BEQ     3$
10025  036756  102001                          BVC     3$
10026  036760  103001                          BCC     4$
10027
10028  036762  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10029
10030  036764  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10031  036766  001401                          BEQ     TST547          ;;BR IF YES
10032
10033  036770  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10034
10035                                  ;;**************************************************************
10036                                  ;*TEST 547      BIS  SM1,DM1 TEST - N:C = 0111
10037                                  ;;**************************************************************
10038  036772                          TST547:
10039  036772  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10040  036774  012700  000547                  MOV     #547,R0         ;;LOAD R0 WITH TEST NUMBER
10041  037000  013701  037030                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10042  037004  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10043  037010  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 1777777
10044  037014  012705  063256                  MOV     #DWTA+10,R5     ;SOURCE ADDR = DWTA+10
10045  037020  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 052525
10046  037024  000257                          CCC                     ;CLEAR FLAGS
10047  037026  000267                          267                     ;N:C = 0111
10048
10049  037030  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10050
10051  037032  100003                          BPL     3$              ;N:C = 1001?
10052  037034  001402                          BEQ     3$
10053  037036  102401                          BVS     3$
10054  037040  103401                          BCS     4$
10055
10056  037042  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10057
10058  037044  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10059  037046  001402                          BEQ     TST550          ;;BR IF YES
10060
10061  037050  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10062  037052  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10063
10064                                  ;;**************************************************************
10065                                  ;*TEST 550      BIS  SM1,DM1 TEST - N:C = 1000
10066                                  ;;**************************************************************
10067  037054                          TST550:
10068  037054  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10069  037056  012700  000550                  MOV     #550,R0         ;;LOAD R0 WITH TEST NUMBER
10070  037062  013701  037106                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10071  037066  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
10072  037072  005004                          CLR     R4              ;RESULT S / B = 000000
10073  037074  012705  063246                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
10074  037100  005012                          CLR     (R2)            ;[DEST] = 000000
10075  037102  000257                          CCC                     ;CLEAR FLAGS
10076  037104  000270                          SEN                     ;N:C = 1000
10077
10078  037106  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10079
10080  037110  100403                          BMI     3$              ;N:C = 0100 ?
10081  037112  001002                          BNE     3$
10082  037114  102401                          BVS     3$
10083  037116  103001                          BCC     4$
10084
10085  037120  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10086
10087  037122  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10088  037124  001402                          BEQ     TST551          ;;BR IF YES
10089
10090  037126  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10091  037130  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10092
10093                                  ;;**********************************************************************
10094                                  ;*TEST 551      BIC  SM1,DM1 TEST - N:C = 0111
10095                                  ;;**********************************************************************
10096  037132                          TST551:
10097  037132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10098  037134  012700  000551                  MOV     #551,R0         ;;LOAD R0 WITH TEST NUMBER
10099  037140  013701  037174                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10100  037144  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10101  037150  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10102  037154  012705  063242                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10103  037160  012715  077777                  MOV     #77777,(R5)     ;[SOURCE] = 77777
10104  037164  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
10105  037170  000257                          CCC                     ;CLEAR FLAGS
10106  037172  000267                          267                     ;N:C = 0111
10107
10108  037174  041512                  2$:     BIC     (R5),(R2)       ;TEST THE BIC
10109
10110  037176  100003                          BPL     3$              ;N:C = 1001 ?
10111  037200  001402                          BEQ     3$
10112  037202  102401                          BVS     3$
10113  037204  103401                          BCS     4$
10114
10115  037206  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10116
10117  037210  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10118  037212  001402                          BEQ     TST552          ;;BR IF YES
10119
10120  037214  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10121  037216  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10122
10123                                  ;;**********************************************************************
10124                                  ;*TEST 552      BIC  SM1,DM1 TEST - N:C = 1000
10125                                  ;;**********************************************************************
10126  037220                          TST552:
10127  037220  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
10128  037222  012700  000552          MOV     #552,R0         ;;LOAD R0 WITH TEST NUMBER
10129  037226  013701  037254          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10130  037232  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10131  037236  005004                  CLR     R4              ;RESULT S / B = 000000
10132  037240  012705  063242          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10133  037244  005015                  CLR     (R5)            ;[SOURCE] = 000000
10134  037246  005012                  CLR     (R2)            ;[DEST] = 000000
10135  037250  000257                  CCC                     ;CLEAR FLAGS
10136  037252  000270                  SEN                     ;N:C = 1000
10137
10138  037254  041512          2$:     BIC     (R5),(R2)       ;TEST THE BIC
10139
10140  037256  100403                  BMI     3$              ;N:C = 0100 ?
10141  037260  001002                  BNE     3$
10142  037262  102401                  BVS     3$
10143  037264  103001                  BCC     4$
10144
10145  037266  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10146
10147  037270  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10148  037272  001402                  BEQ     TST553          ;;BR IF YES
10149
10150  037274  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10151  037276  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10152
10153                          ;;**************************************************************
10154                          ;*TEST 553      BIT  SM1,DM1 TEST - N:C = 1000
10155                          ;**************************************************************
10156  037300                  TST553:
10157  037300  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10158  037302  012700  000553          MOV     #553,R0         ;;LOAD R0 WITH TEST NUMBER
10159  037306  013701  037342          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10160  037312  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10161  037316  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
10162  037322  012705  063242          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10163  037326  012715  052525          MOV     #52525,(R5)     ;[SOURCE] = 052525
10164  037332  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
10165  037336  000257                  CCC                     ;CLEAR FLAGS
10166  037340  000270                  SEN                     ;N:C = 1000
10167
10168  037342  031512          2$:     BIT     (R5),(R2)       ;TEST THE BIT
10169
10170  037344  100403                  BMI     3$              ;N:C = 0100 ?
10171  037346  001002                  BNE     3$
10172  037350  102401                  BVS     3$
10173  037352  103001                  BCC     4$
10174
10175  037354  104001          3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10176
10177  037356  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10178  037360  001402                  BEQ     TST554          ;;BR IF YES
10179
10180  037362  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10181  037364  104001          5$:     ERROR   1               ;BIT DELIVERED A RESULT
10182
10183                          ;;**************************************************************
```

```
10184                                   ;*TEST 554      BIT  SM1,DM1 TEST - N:C = 0111
10185                                   ;;*****************************************************************
10186  037366                           TST554:
10187  037366  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10188  037370  012700  000554                   MOV     #554,R0         ;;LOAD R0 WITH TEST NUMBER
10189  037374  013701  037442                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10190                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT11
10191  037400  032737  004000  063160           BIT     #BIT11,@#BPTLOC ;BREAKPOINT HALT SET ??
10192  037406  001401                           BEQ     .+4             ;BR IF NOT
10193  037410  000000                           HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10194  037412  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10195  037416  012704  100000                   MOV     #100000,R4      ;RESULT S / B = 100000
10196  037422  012705  063242                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10197  037426  012715  100000                   MOV     #100000,(R5)    ;[SOURCE] = 100000
10198  037432  012712  100000                   MOV     #100000,(R2)    ;[DEST] = 100000
10199  037436  000257                           CCC                     ;CLEAR FLAGS
10200  037440  000267                           267                     ;N:C = 0111
10201
10202  037442  031512                   2$:     BIT     (R5),(R2)       ;TEST THE BIT
10203
10204  037444  100003                           BPL     3$              ;N:C = 1001 ?
10205  037446  001402                           BEQ     3$
10206  037450  102401                           BVS     3$
10207  037452  103401                           BCS     4$
10208
10209  037454  104001                   3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10210
10211  037456  020412                   4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10212  037460  001402                           BEQ     TST555          ;;BR IF YES
10213
10214  037462  011203                           MOV     (R2),R3         ;GET THE WAS DATA
10215  037464  104001                   5$:     ERROR   1               ;BIT DELIVERED A RESULT
10216
10217                                   ;;*****************************************************************
10218                                   ;*TEST 555      CMP  SM1,DM1 TEST - N:C = 1010
10219                                   ;;*****************************************************************
10220  037466                           TST555:
10221  037466  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10222  037470  012700  000555                   MOV     #555,R0         ;;LOAD R0 WITH TEST NUMBER
10223  037474  013701  037526                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10224  037500  012702  063236                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10225  037504  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
10226  037510  012705  063242                   MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10227  037514  012715  177777                   MOV     #-1,(R5)        ;[SOURCE] = 177777
10228  037520  010412                           MOV     R4,(R2)         ;[DEST] = 177777
10229  037522  000257                           CCC                     ;CLEAR FLAGS
10230  037524  000272                           272                     ;N:C = 1010
10231
10232  037526  021512                   2$:     CMP     (R5),(R2)       ;TEST THE CMP
10233
10234  037530  100403                           BMI     3$              ;N:C = 0100 ?
10235  037532  001002                           BNE     3$
10236  037534  102401                           BVS     3$
10237  037536  103001                           BCC     4$
10238
10239  037540  104001                   3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
```

```
10240
10241  037542  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10242  037544  001402                          BEQ     TST556          ;;BR IF YES
10243
10244  037546  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10245  037550  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10246
10247                                  ;;**********************************************************************
10248                                  ;*TEST 556       CMP  SM1,DM1 TEST - N:C = 0110
10249                                  ;;**********************************************************************
10250  037552                          TST556:
10251  037552  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10252  037554  012700  000556                  MOV     #556,R0         ;;LOAD R0 WITH TEST NUMBER
10253  037560  013701  037612                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10254  037564  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10255  037570  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
10256  037574  012705  063242                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10257  037600  005015                          CLR     (R5)            ;[SOURCE] = 000000
10258  037602  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
10259  037606  000257                          CCC                     ;CLEAR FLAGS
10260  037610  000266                          266                     ;N:C = 0110
10261
10262  037612  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
10263
10264  037614  100003                          BPL     3$              ;N:C = 1001 ?
10265  037616  001402                          BEQ     3$
10266  037620  102401                          BVS     3$
10267  037622  103401                          BCS     4$
10268
10269  037624  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10270
10271  037626  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10272  037630  001402                          BEQ     TST557          ;;BR IF YES
10273
10274  037632  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10275  037634  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10276
10277                                  ;;**********************************************************************
10278                                  ;*TEST 557       CMP  SM1,DM1 TEST - N:C = 0000
10279                                  ;;**********************************************************************
10280  037636                          TST557:
10281  037636  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10282  037640  012700  000557                  MOV     #557,R0         ;;LOAD R0 WITH TEST NUMBER
10283  037644  013701  037676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10284  037650  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10285  037654  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
10286  037660  012705  063242                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10287  037664  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 000000
10288  037670  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
10289  037674  000257                          CCC                     ;CLEAR FLAGS
10290
10291  037676  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
10292
10293  037700  100403                          BMI     3$              ;N:C = 0010 ?
10294  037702  001402                          BEQ     3$
10295  037704  102001                          BVC     3$
```

```
10296  037706  103001                          BCC     4$
10297
10298  037710  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10299
10300  037712  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10301  037714  001402                          BEQ     TST560          ;;BR IF YES
10302
10303  037716  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10304  037720  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10305
10306                                  ;;**********************************************************************
10307                                  ;*TEST 560      BISB  SM1,DM0 TEST - SOURCE ADDR 000
10308                                  ;;**********************************************************************
10309  037722                          TST560:
10310  037722  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10311  037724  012700  000560                  MOV     #560,R0         ;;LOAD R0 WITH TEST NUMBER
10312  037730  013701  037750                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10313  037734  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
10314  037740  012705  064555                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10315  037744  005003                          CLR     R3              ;[DEST] = 000000
10316  037746  000257                          CCC                     ;SCOPE SYNC
10317
10318  037750  151503                  2$:     BISB    (R5),R3         ;TEST THE BISB
10319
10320  037752  020403                          CMP     R4,R3           ;RESULT CORRECT ?
10321  037754  001401                          BEQ     TST561          ;;BR IF YES
10322
10323  037756  104002                  3$:     ERROR   2               ;BISB DELIVERED THE WRONG RESULT
10324
10325                                  ;;**********************************************************************
10326                                  ;*TEST 561      BISB  SM1,DM1 TEST - SOURCE ADDR 000
10327                                  ;;**********************************************************************
10328  037760                          TST561:
10329  037760  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10330  037762  012700  000561                  MOV     #561,R0         ;;LOAD R0 WITH TEST NUMBER
10331  037766  013701  040012                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10332  037772  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10333  037776  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
10334  040002  012705  064555                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10335  040006  005012                          CLR     (R2)            ;[DEST] = 000000
10336  040010  000257                          CCC                     ;SCOPE SYNC
10337
10338  040012  151512                  2$:     BISB    (R5),(R2)       ;TEST THE BISB
10339
10340  040014  020412                          CMP     R4,(R2)         ;CORRECT RESULT
10341  040016  001402                          BEQ     TST562          ;;BR IF YES
10342
10343  040020  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10344  040022  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10345
10346                                  ;;**********************************************************************
10347                                  ;*TEST 562      BISB  SM1,DM2 TEST - SOURCE ADDR 000
10348                                  ;;**********************************************************************
10349  040024                          TST562:
10350  040024  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10351  040026  012700  000562                  MOV     #562,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
MAINDEC-11-DQKDA-A   KD11-K BASIC LOGIC TESTS   MACY11 27(1006)  08-FEB-77  16:23  PAGE 189
DQKDAA.P11    08-FEB-77 16:17                   T562    BISB  SM1,DM2 TEST - SOURCE ADDR ODD

 10352  040032  013701  040060                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 10353  040036  012702  063236                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 10354  040042  012704  000377                          MOV     #377,R4         ;RESULT S / B = 377
 10355  040046  012705  064555                          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 10356  040052  005012                                  CLR     (R2)            ;[DEST] = 000000
 10357  040054  010203                                  MOV     R2,R3           ;DEST ADDR IN R3
 10358  040056  000257                                  CCC                     ;SCOPE SYNC
 10359
 10360  040060  151523                          2$:     BISB    (R5),(R3)+      ;TEST THE BISB
 10361
 10362  040062  020412                                  CMP     R4,(R2)         ;CORRECT RESULT
 10363  040064  001402                                  BEQ     TST563          ;;BR IF YES
 10364
 10365  040066  011203                                  MOV     (R2),R3         ;GET THE WAS DATA
 10366  040070  104001                          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
 10367
 10368                                          ;;**************************************************************
 10369                                          ;*TEST 563      BISB  SM1,DM3 TEST - SOURCE ADDR ODD
 10370                                          ;;**************************************************************
 10371  040072                                  TST563:
 10372  040072  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 10373  040074  012700  000563                          MOV     #563,R0         ;;LOAD R0 WITH TEST NUMBER
 10374  040100  013701  040130                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 10375  040104  012702  063236                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 10376  040110  012704  000377                          MOV     #377,R4         ;RESULT S / B = 377
 10377  040114  012705  064555                          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 10378  040120  005012                                  CLR     (R2)            ;[DEST] = 000000
 10379  040122  012703  063232                          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 10380  040126  000257                                  CCC                     ;SCOPE SYNC
 10381
 10382  040130  151533                          2$:     BISB    (R5),@(R3)+     ;TEST THE BISB
 10383
 10384  040132  020412                                  CMP     R4,(R2)         ;CORRECT RESULT
 10385  040134  001402                                  BEQ     TST564          ;;BR IF YES
 10386
 10387  040136  011203                                  MOV     (R2),R3         ;GET THE WAS DATA
 10388  040140  104001                          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
 10389
 10390                                          ;;**************************************************************
 10391                                          ;*TEST 564      BISB  SM1,DM4 TEST - SOURCE ADDR ODD
 10392                                          ;;**************************************************************
 10393  040142                                  TST564:
 10394  040142  000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 10395  040144  012700  000564                          MOV     #564,R0         ;;LOAD R0 WITH TEST NUMBER
 10396  040150  013701  040200                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 10397  040154  012702  063236                          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 10398  040160  012704  177400                          MOV     #177400,R4      ;RESULT S / B = 177400
 10399  040164  012705  064555                          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 10400  040170  012703  063240                          MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
 10401  040174  005012                                  CLR     (R2)            ;[DEST] = 000000
 10402  040176  000257                                  CCC                     ;SCOPE SYNC
 10403
 10404  040200  151543                          2$:     BISB    (R5),-(R3)      ;TEST THE BISB
 10405
 10406  040202  020412                                  CMP     R4,(R2)         ;CORRECT RESULT
 10407  040204  001402                                  BEQ     TST565          ;;BR IF YES
```

```
10408
10409  040206  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10410  040210  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10411
10412                                  ;;*****************************************************************
10413                                  ;*TEST 565      BISB  SM1,DM5 TEST - SOURCE ADDR ODD
10414                                  ;;*****************************************************************
10415  040212                          TST565:
10416  040212  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10417  040214  012700  000565                  MOV     #565,R0         ;;LOAD R0 WITH TEST NUMBER
10418  040220  013701  040250                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10419  040224  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10420  040230  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
10421  040234  012705  064555                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10422  040240  012703  063234                  MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
10423  040244  005012                          CLR     (R2)            ;[DEST] = 000000
10424  040246  000257                          CCC                     ;SCOPE SYNC
10425
10426  040250  151553                  2$:     BISB    (R5),@-(R3)     ;TEST THE BISB
10427
10428  040252  020412                          CMP     R4,(R2)         ;CORRECT RESULT
10429  040254  001402                          BEQ     TST566          ;;BR IF YES
10430
10431  040256  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10432  040260  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10433
10434                                  ;;*****************************************************************
10435                                  ;*TEST 566      BISB  SM1,DM6 TEST - SOURCE ADDR ODD
10436                                  ;;*****************************************************************
10437  040262                          TST566:
10438  040262  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10439  040264  012700  000566                  MOV     #566,R0         ;;LOAD R0 WITH TEST NUMBER
10440  040270  013701  040320                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10441  040274  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10442  040300  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
10443  040304  012705  064555                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10444  040310  012703  063244                  MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
10445  040314  005012                          CLR     (R2)            ;[DEST] = 000000
10446  040316  000257                          CCC                     ;SCOPE SYNC
10447
10448  040320  151563  177772          2$:     BISB    (R5),-6(R3)     ;TEST THE BISB
10449
10450  040324  020412                          CMP     R4,(R2)         ;CORRECT RESULT
10451  040326  001402                          BEQ     TST567          ;;BR IF YES
10452
10453  040330  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10454  040332  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10455
10456                                  ;;*****************************************************************
10457                                  ;*TEST 567      BISB  SM1,DM7 TEST - SOURCE ADDR ODD
10458                                  ;;*****************************************************************
10459  040334                          TST567:
10460  040334  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10461  040336  012700  000567                  MOV     #567,R0         ;;LOAD R0 WITH TEST NUMBER
10462  040342  013701  040372                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10463  040346  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
```

```
10464  040352  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
10465  040356  012705  064555          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10466  040362  012703  063222          MOV     #ATA,R3         ;BASE DEST ADDR = ATA
10467  040366  005012                  CLR     (R2)            ;[DEST] = 000000
10468  040370  000257                  CCC                     ;SCOPE SYNC
10469
10470  040372  151573  000010  2$:     BISB    (R5),@10(R3)    ;TEST THE BISB
10471
10472  040376  020412                  CMP     R4,(R2)         ;CORRECT RESULT
10473  040400  001402                  BEQ     TST570          ;;BR IF YES
10474
10475  040402  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10476  040404  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10477
10478                          ;;*****************************************************************
10479                          ;*TEST 570      BISB  SM0,DM2 TEST - DEST ADDR EVEN
10480                          ;;*****************************************************************
10481  040406                  TST570:
10482  040406  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10483  040410  012700  000570          MOV     #570,R0         ;;LOAD R0 WITH TEST NUMBER
10484  040414  013701  040436          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10485  040420  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10486  040424  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
10487  040430  010203                  MOV     R2,R3           ;DEST ADDR IN R3
10488  040432  005012                  CLR     (R2)            ;[DEST] = 000000
10489  040434  000257                  CCC                     ;SCOPE SYNC
10490
10491  040436  150423          2$:     BISB    R4,(R3)+        ;TEST THE BISB
10492
10493  040440  020412                  CMP     R4,(R2)         ;CORRECT RESULT
10494  040442  001402                  BEQ     TST571          ;;BR IF YES
10495
10496  040444  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10497  040446  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10498
10499                          ;;*****************************************************************
10500                          ;*TEST 571      BISB  SM0,DM1 TEST - DEST ADDR ODD
10501                          ;;*****************************************************************
10502  040450                  TST571:
10503  040450  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10504  040452  012700  000571          MOV     #571,R0         ;;LOAD R0 WITH TEST NUMBER
10505  040456  013701  040506          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10506  040462  012702  063236          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10507  040466  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
10508  040472  012705  000377          MOV     #377,R5         ;[R5]=SOURCE OPR = 377
10509  040476  012703  063237          MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
10510  040502  005012                  CLR     (R2)            ;[DEST] = 000000
10511  040504  000257                  CCC                     ;SCOPE SYNC
10512
10513  040506  150513          2$:     BISB    R5,(R3)         ;TEST THE BISB
10514
10515  040510  020412                  CMP     R4,(R2)         ;CORRECT RESULT
10516  040512  001402                  BEQ     TST572          ;;BR IF YES
10517
10518  040514  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10519  040516  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
```

```
10520
10521                                   ;;*****************************************************************
10522                                   ;*TEST 572      BISB  SM0,DM1 TEST - DEST ADDR EVEN
10523                                   ;;*****************************************************************
10524  040520                           TST572:
10525  040520  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10526  040522  012700  000572                  MOV     #572,R0         ;;LOAD R0 WITH TEST NUMBER
10527  040526  013701  040550                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10528  040532  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10529  040536  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
10530  040542  010203                          MOV     R2,R3           ;DEST ADDR IN R3
10531  040544  005012                          CLR     (R2)            ;[DEST] = 000000
10532  040546  000257                          CCC                     ;SCOPE SYNC
10533
10534  040550  150413                  2$:     BISB    R4,(R3)         ;TEST THE BISB
10535
10536  040552  020412                          CMP     R4,(R2)         ;CORRECT RESULT
10537  040554  001402                          BEQ     TST573          ;;BR IF YES
10538
10539  040556  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10540  040560  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10541
10542                                   ;;*****************************************************************
10543                                   ;*TEST 573      BISB  SM1,DM1 TEST - DEST ADDR ODD
10544                                   ;;*****************************************************************
10545  040562                           TST573:
10546  040562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10547  040564  012700  000573                  MOV     #573,R0         ;;LOAD R0 WITH TEST NUMBER
10548  040570  013701  040620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10549  040574  012702  063236                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10550  040600  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
10551  040604  012705  064555                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
10552  040610  012703  063237                  MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
10553  040614  005012                          CLR     (R2)            ;[DEST] = 000000
10554  040616  000257                          CCC                     ;SCOPE SYNC
10555
10556  040620  151513                  2$:     BISB    (R5),(R3)       ;TEST THE BISB
10557
10558  040622  020412                          CMP     R4,(R2)         ;CORRECT RESULT
10559  040624  001402                          BEQ     TST574          ;;BR IF YES
10560
10561  040626  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10562  040630  104001                  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
10563
10564                                   ;;*****************************************************************
10565                                   ;*TEST 574      JMP MODE 1 TEST, FLAGS = 1111
10566                                   ;;*****************************************************************
10567  040632                           TST574:
10568  040632  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10569  040634  012700  000574                  MOV     #574,R0         ;;LOAD R0 WITH TEST NUMBER
10570  040640  013701  040652                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10571  040644  012702  040660                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
10572  040650  000277                          SCC                     ;MAKE N:C = 1111
10573
10574  040652  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
10575
```

```
10576  040654  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10577  040656  000405                          BR      TST575          ;;GO CALL SCOPE
10578
10579  040660  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
10580  040662  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
10581  040664  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
10582  040666  100401                          BMI     TST575          ;;BR IF "N" STILL SET
10583
10584  040670  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED ONE
10585
10586                                  ;;**********************************************************************
10587                                  ;*TEST 575       JMP MODE 1 TEST, FLAGS = 0000
10588                                  ;;**********************************************************************
10589  040672                          TST575:
10590  040672  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10591  040674  012700  000575                  MOV     #575,R0         ;;LOAD R0 WITH TEST NUMBER
10592  040700  013701  040712                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10593  040704  012702  040720                  MOV     #4$,R2          ;R2 CONTAINS JMP ADDRESS
10594  040710  000257                          CCC                     ;MAKE N:C = 0000
10595
10596  040712  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
10597
10598  040714  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10599  040716  000405                          BR      TST576          ;;GO CALL SCOPE
10600
10601  040720  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
10602  040722  102402                          BVS     5$              ;BR IF JMP SET "V"
10603  040724  001401                          BEQ     5$              ;BR IF JMP SET "Z"
10604  040726  100001                          BPL     TST576          ;;BR IF "N" STILL CLEAR
10605
10606  040730  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET ONE
10607
10608                                  ;;**********************************************************************
10609                                  ;*TEST 576       JMP MODE 2 TEST; FLAGS = 1111
10610                                  ;;**********************************************************************
10611  040732                          TST576:
10612  040732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10613  040734  012700  000576                  MOV     #576,R0         ;;LOAD R0 WITH TEST NUMBER
10614  040740  013701  040752                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10615  040744  012702  040760                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
10616  040750  000277                          SCC                     ;SET N:C = 1111
10617
10618  040752  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
10619
10620  040754  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10621  040756  000411                          BR      TST577          ;;GO TO SCOPE EXIT
10622
10623  040760  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
10624  040762  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
10625  040764  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
10626  040766  100401                          BMI     6$              ;BR IF "N" STILL SET
10627
10628  040770  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
10629
10630  040772  022702  040762          6$:     CMP     #4$+2,R2        ;DID R2 GET AUTO-INCREMENTED?
10631  040776  001401                          BEQ     TST577          ;;BR IF YES
```

```
10632
10633  041000  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER (R2)
10634
10635                                  ;;**********************************************************************
10636                                  ;*TEST 577       JMP MODE 2 TEST; FLAGS = 0000
10637                                  ;;**********************************************************************
10638  041002                          TST577:
10639  041002  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10640  041004  012700  000577                 MOV     #577,R0         ;;LOAD R0 WITH TEST NUMBER
10641  041010  013701  041022                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10642  041014  012702  041030                 MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
10643  041020  000257                         CCC                     ;MAKE N:C = 0000
10644
10645  041022  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
10646
10647  041024  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10648  041026  000405                         BR      TST600          ;;GO TO SCOPE EXIT
10649
10650  041030  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
10651  041032  102402                          BVS     5$              ;BR IF JMP SET "V"
10652  041034  001401                          BEQ     5$              ;BR IF JMP SET "Z"
10653  041036  100001                          BPL     TST600          ;;BR IF "N" IS CLEAR
10654
10655  041040  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
10656
10657                                  ;;**********************************************************************
10658                                  ;*TEST 600       JMP TEST MODE 3; FLAGS = 1111
10659                                  ;;**********************************************************************
10660  041042                          TST600:
10661  041042  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10662  041044  012700  000600                 MOV     #600,R0         ;;LOAD R0 WITH TEST NUMBER
10663  041050  013701  041062                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10664  041054  012702  041114                 MOV     #7$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
10665  041060  000277                         SCC                     ;SET N:C = 1111
10666
10667  041062  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
10668
10669  041064  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10670  041066  000414                         BR      TST601          ;;GO TO SCOPE EXIT
10671
10672  041070  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
10673  041072  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
10674  041074  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
10675  041076  100401                          BMI     6$              ;BR IF "N" STILL SET
10676
10677  041100  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEAR
10678
10679  041102  022702  041116          6$:     CMP     #7$+2,R2        ;DID JMP UPDATE R2?
10680  041106  001404                          BEQ     TST601          ;;BR IF YES
10681
10682  041110  104006                          ERROR   6               ;JMP FAILED TO UPDATE REGISTER
10683  041112  000402                          BR      TST601          ;;GO TO SCOPE EXIT
10684  041114  041070                  7$:     4$                      ;JMP3 CONTAINS JUMP ADDRESS
10685  041116  104006                          ERROR   6               ;ERROR CALL OCCURS IF MODE3 HAPPENS
10686                                                                  ;TO EXECUTE AS MODE 1 OR 2 AND
10687                                                                  ;4$ IS LEGAL INSTRUCTION
```

```
10688
10689                                   ;;*****************************************************************
10690                                   ;*TEST 601      JMP TEST MODE 3; FLAGS = 0000
10691                                   ;;*****************************************************************
10692  041120                           TST601:
10693  041120  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
10694  041122  012700  000601                  MOV     #601,R0                 ;;LOAD R0 WITH TEST NUMBER
10695  041126  013701  041140                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
10696  041132  012702  041162                  MOV     #6$,R2                  ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
10697  041136  000257                          CCC                             ;MAKE N:C = 0000
10698
10699  041140  000132                  2$:     JMP     @(R2)+                  ;TEST THE JMP - GO TO 4$
10700
10701  041142  104006                  3$:     ERROR   6                       ;JMP FAILED TO LOAD THE PC
10702  041144  000410                          BR      TST602                  ;;GO TO SCOPE EXIT
10703
10704  041146  103403                  4$:     BCS     5$                      ;BR IF JMP SET "C"
10705  041150  102402                          BVS     5$                      ;BR IF JMP SET "V"
10706  041152  001401                          BEQ     5$                      ;BR IF JMP SET "Z"
10707  041154  100004                          BPL     TST602                  ;;BR IF "N" STILL CLEAR
10708
10709  041156  104006                  5$:     ERROR   6                       ;JMP ALTERED CODES - SET
10710  041160  000402                          BR      TST602                  ;;GO TO SCOPE EXIT
10711
10712  041162  041146                  6$:     4$                              ;JUMP ADDRESS IN 6$
10713  041164  104006                          ERROR   6                       ;JMP MODE 3 EXECUTED LIKE MODE 1 OR 2
10714
10715                                   ;;*****************************************************************
10716                                   ;*TEST 602      JMP TEST MODE 4; FLAGS = 1111
10717                                   ;;*****************************************************************
10718  041166                           TST602:
10719  041166  000004                          SCOPE                           ;CALL THE SCOPE LOOP UTILITY
10720  041170  012700  000602                  MOV     #602,R0                 ;;LOAD R0 WITH TEST NUMBER
10721  041174  013701  041206                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
10722  041200  012702  041216                  MOV     #5$,R2  ;[R2] = JMP ADDRESS PLUS 2
10723  041204  000277                          SCC                             ;MAKE N:C = 1111
10724
10725  041206  000142                  2$:     JMP     -(R2)                   ;TEST THE JMP - GO TO 5$ MINUS 2
10726
10727  041210  104006                  3$:     ERROR   6                       ;JMP FAILED TO LOAD PC
10728  041212  000414                          BR      TST603                  ;;GO TO SCOPE EXIT
10729
10730  041214  000402                          BR      4$                      ;GO TEST FLAGS - JMP LOADED PC OK
10731  041216  104006                  5$:     ERROR   6                       ;JMP FAILED TO AUTO-DECREMENT R2
10732  041220  000411                          BR      TST603                  ;;GO TO SCOPE EXIT
10733
10734  041222  103003                  4$:     BCC     7$                      ;BR IF JMP CLEARED "C"
10735  041224  102002                          BVC     7$                      ;BR IF JMP CLEARED "V"
10736  041226  001001                          BNE     7$                      ;BR IF JMP CLEARED "Z"
10737  041230  100401                          BMI     6$                      ;BR IF "N" STILL SET
10738
10739  041232  104006                  7$:     ERROR   6                       ;JMP ALTERED FLAGS
10740
10741  041234  022702  041214          6$:     CMP     #5$-2,R2                ;DID JMP UPDATE R2 PROPERLY?
10742  041240  001401                          BEQ     TST603                  ;;BR IF YES
10743
```

```
10744  041242  104006                  9$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
10745
10746                                  ;;**********************************************************************
10747                                  ;*TEST 603      JMP TEST MODE 4; FLAGS = 0000
10748                                  ;;**********************************************************************
10749  041244                          TST603:
10750  041244  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10751  041246  012700  000603                 MOV     #603,R0         ;;LOAD R0 WITH TEST NUMBER
10752  041252  013701  041264                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10753  041256  012702  041274                 MOV     #4$+2,R2        ;[R2] = JUMP ADDRESS PLUS 2
10754  041262  000257                         CCC                     ;MAKE N:C = 0000
10755
10756  041264  000142                  2$:     JMP     -(R2)           ;TEST THE JMP - TO TO 4$
10757
10758  041266  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10759  041270  000405                          BR      TST604          ;;GO TO SCOPE EXIT
10760
10761  041272  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
10762  041274  102402                          BVS     5$              ;BR IF JMP SET "V"
10763  041276  001401                          BEQ     5$              ;BR IF JMP SET "Z"
10764  041300  100001                          BPL     TST604          ;;BR IF "N" STILL CLEAR
10765
10766  041302  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
10767
10768                                  ;;**********************************************************************
10769                                  ;*TEST 604      JMP TEST MODE 5; FLAGS = 1111
10770                                  ;;**********************************************************************
10771  041304                          TST604:
10772  041304  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10773  041306  012700  000604                 MOV     #604,R0         ;;LOAD R0 WITH TEST NUMBER
10774  041312  013701  041324                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10775  041316  012702  041360                 MOV     #JMP5,R2        ;JMP CONTAINS ADDR+2 OF JUMP ADDRESS
10776  041322  000277                         SCC
10777
10778  041324  000152                  2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
10779
10780  041326  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10781  041330  000414                          BR      TST605          ;;GO TO SCOPE OXIT
10782
10783  041332  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
10784  041334  102002                          BVC     5$
10785  041336  001001                          BNE     5$
10786  041340  100401                          BMI     6$
10787
10788  041342  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
10789
10790  041344  022702  041356          6$:     CMP     #JMP5-2,R2      ;DID R2 GET AUTO-DECREMENTED
10791  041350  001404                          BEQ     TST605          ;;BR IF YES
10792
10793  041352  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
10794  041354  000402                          BR      TST605          ;;GO TO SCOPE EXIT
10795  041356  041332                          4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
10796  041360  104006                  JMP5:   ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
10797
10798                                  ;;**********************************************************************
10799                                  ;*TEST 605      JMP TEST MODE 5; FLAG = 0000
```

```
10800                                   ;;*****************************************************************
10801  041362                           TST605:
10802  041362  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10803  041364  012700  000605                   MOV     #605,R0         ;;LOAD R0 WITH TEST NUMBER
10804  041370  013701  041402                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10805  041374  012702  041426                   MOV     #JMPSA,R2       ;[R2] = ADDR +2 OF JUMP ADDRESS
10806  041400  000257                           CCC                     ;SET N:C = 0000
10807
10808  041402  000152                   2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
10809
10810  041404  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10811  041406  000410                           BR      TST606          ;;GO TO SCOPE EXIT
10812
10813  041410  103403                   4$:     BCS     5$              ;BR IF JMP SET "C"
10814  041412  102402                           BVS     5$              ;BR IF JMP SET "V"
10815  041414  001401                           BEQ     5$              ;BR IF JMP SET "Z"
10816  041416  100004                           BPL     TST606          ;;BR IF "N" STILL CLEAR
10817
10818  041420  104006                   5$:     ERROR   6               ;JMP ALTERED THE CODES - SET
10819  041422  000402                           BR      TST606          ;;GO TO SCOPE EXIT
10820
10821  041424  041410                           4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
10822  041426  104006                   JMPSA:  ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
10823
10824                                   ;;*****************************************************************
10825                                   ;*TEST 606      JMP TEST MODE 6; FLAGS = 1111
10826                                   ;;*****************************************************************
10827  041430                           TST606:
10828  041430  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10829  041432  012700  000606                   MOV     #606,R0         ;;LOAD R0 WITH TEST NUMBER
10830  041436  013701  041450                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10831  041442  012702  041474                   MOV     #7$,R2  ;[R2] = BASE ADDRESS TO BE INDEXED
10832  041446  000277                           SCC                     ;MAKE N:C = 1111
10833
10834  041450  000162  177764           2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
10835
10836  041454  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
10837  041456  000407                           BR      TST607          ;;GO TO SCOPE EXIT
10838
10839  041460  103003                   4$:     BCC     5$              ;BR IF JMP CLEARED "C"
10840  041462  102002                           BVC     5$
10841  041464  001001                           BNE     5$
10842  041466  100403                           BMI     TST607          ;;BR IF "N" STILL SET
10843
10844  041470  104006                   5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
10845  041472  000401                           BR      TST607          ;;GO TO SCOPE EXIT
10846
10847  041474  104006                   7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2 OR
10848                                                                   ;FAILED TO INDEX [R2]
10849
10850                                   ;;*****************************************************************
10851                                   ;*TEST 607      JMP TEST MODE 6; FLAGS = 0000
10852                                   ;;*****************************************************************
10853  041476                           TST607:
10854  041476  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10855  041500  012700  000607                   MOV     #607,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
10856  041504  013701  041516                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10857  041510  012702  041542                  MOV     #7$,R2  ;[R2] = BASE ADDRESS FOR JUMP
10858  041514  000257                          CCC                     ;MAKE N:C = 0000
10859
10860  041516  000162  177764          2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
10861
10862  041522  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10863  041524  000407                          BR      TST610          ;;GO TO SCOPE EXIT
10864
10865  041526  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
10866  041530  102402                          BVS     5$              ;BR IF JMP SET "V"
10867  041532  001401                          BEQ     5$              ;BR IF JMP SET "Z"
10868  041534  100003                          BPL     TST610          ;;BR IF "N" STILL CLEAR
10869
10870  041536  104006                  5$:     ERROR   6               ;JMP ALTERED CODES
10871  041540  000401                          BR      TST610          ;;GO TO SCOPE EXIT
10872
10873  041542  104006                  7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2, OR
10874                                                                  ;FAILED TO INDEX [R2]
10875
10876                                  ;;**************************************************************
10877                                  ;*TEST 610      JMP TEST MODE 7; FLAGS = 1111
10878                                  ;;**************************************************************
10879  041544                          TST610:
10880  041544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10881  041546  012700  000610                  MOV     #610,R0         ;;LOAD R0 WITH TEST NUMBER
10882  041552  013701  041564                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10883  041556  012702  041574                  MOV     #5$,R2  ;[R2] = BASE ADDRESS
10884  041562  000277                          SCC                     ;MAKE N:C = 1111
10885
10886  041564  000172  000020          2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
10887
10888  041570  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10889  041572  000412                          BR      TST611          ;;GO TO SCOPE EXIT
10890
10891  041574  104006                  5$:     ERROR   6               ;JMP FAILED TO INDEX OR ACTED LIKE MODE 1 OR 2
10892  041576  000410                          BR      TST611          ;;GO TO SCOPE EXIT
10893
10894  041600  103003                  4$:     BCC     7$              ;BR IF JMP CLEARED "C"
10895  041602  102002                          BVC     7$              ;BR IF JMP CLEARED "V"
10896  041604  001001                          BNE     7$              ;BR IF JMP CLEARED "Z"
10897  041606  100404                          BMI     TST611          ;;BR IF "N" STILL SET
10898
10899  041610  104006                  7$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
10900  041612  000402                          BR      TST611          ;;GO TO SCOPE EXIT
10901
10902  041614  041600                  8$:     4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
10903
10904  041616  104006                          ERROR   6               ;JMP EXECUTED LIKE MODE 6
10905
10906                                  ;;**************************************************************
10907                                  ;*TEST 611      JMP TEST MODE 7; FLAGS = 0000
10908                                  ;;**************************************************************
10909  041620                          TST611:
10910  041620  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10911  041622  012700  000611                  MOV     #611,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
10912  041626  013701  041640          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10913  041632  012702  041650          MOV     #5$,R2  ;[R2] = BASE ADDRESS
10914  041636  000257                  CCC                     ;MAKE N:C = 0000
10915
10916  041640  000172  000020  2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
10917
10918  041644  104006          3$:     ERROR   6               ;JMP FAILED TO LOAD PC
10919  041646  000412                  BR      TST612          ;;GO TO SCOPE EXIT
10920
10921  041650  104006          5$:     ERROR   6               ;JMP FAILED TO INDEX
10922  041652  000410                  BR      TST612          ;;GO TO SCOPE EXIT
10923
10924  041654  103403          4$:     BCS     7$              ;BR IF JMP SET "C"
10925  041656  102402                  BVS     7$              ;BR IF JMP SET "V"
10926  041660  001401                  BEQ     7$              ;BR IF JMP SET "Z"
10927  041662  100004                  BPL     TST612          ;;BR IF "N" STILL CLEAR
10928
10929  041664  104006          7$:     ERROR   6               ;JMP ALTERED CODES - SET
10930  041666  000402                  BR      TST612          ;;GO TO SCOPE EXIT
10931
10932  041670  041654          8$:     4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
10933
10934  041672  104006                  ERROR   6               ;JMP EXECUTED LIKE A MODE 6
10935
10936                           ;;**************************************************************
10937                           ;*TEST 612      JSR MODE 1 TEST - LOAD PC  /  PUSH SP
10938                           ;;**************************************************************
10939  041674                  TST612:
10940  041674  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10941  041676  012700  000612          MOV     #612,R0         ;;LOAD R0 WITH TEST NUMBER
10942  041702  013701  041724          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10943  041706  010605                  MOV     SP,R5           ;SAVE THE SP
10944  041710  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
10945  041714  010506          1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
10946  041716  012702  041730          MOV     #4$,R2          ;DEST ADDR = 4$
10947  041722  000257                  CCC                     ;SCOPE SYNC
10948
10949  041724  004412          2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
10950
10951  041726  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
10952
10953  041730  005726          4$:     TST     (SP)+           ;POP THE SP
10954  041732  020605                  CMP     SP,R5           ;DID JSR PUSH THE SP ?
10955  041734  001406                  BEQ     TST613          ;;BR IF YES
10956
10957  041736  005746                  TST     -(SP)           ;RESTORE ERROR SP
10958  041740  010603                  MOV     SP,R3           ;[R3]= WAS SP
10959  041742  010504                  MOV     R5,R4
10960  041744  005744                  TST     -(R4)           ;[R4]= S/B SP
10961  041746  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH THE SP
10962
10963  041750  010506                  MOV     R5,SP           ;RESTORE SP IN CASE OF ERROR
10964                           ;;**************************************************************
10965                           ;*TEST 613      JSR MODE 1 TEST - CHECK RN AND OLD PC
10966                           ;;**************************************************************
10967  041752                  TST613:
```

```
10968  041752  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10969  041754  012700  000613                  MOV     #613,R0         ;;LOAD R0 WITH TEST NUMBER
10970  041760  013701  042012                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10971  041764  010605                          MOV     SP,R5           ;SAVE THE SP
10972  041766  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
10973  041772  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
10974  041774  012702  042016                  MOV     #4$,R2          ;DEST ADDR = 4$
10975  042000  005066  177776                  CLR     -2(SP)          ;INIT STACK LOC TO GET [R4]
10976  042004  012704  125252                  MOV     #125252,R4      ;INIT RN = 125252
10977  042010  000257                          CCC                     ;SCOPE SYNC
10978
10979  042012  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
10980
10981  042014  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
10982
10983  042016  022726  125252          4$:     CMP     #125252,(SP)+   ;DID JSR SAVE REG ON STACK
10984  042022  001401                          BEQ     8$              ;BR IF IT DID
10985
10986  042024  104005                  5$:     ERROR   5               ;JSR FAILED TO SAVE REG ON STACK
10987
10988  042026  022704  042014          8$:     CMP     #3$,R4          ;DID OLD PC GET SAVED ?
10989  042032  001401                          BEQ     6$              ;BR IF YES
10990
10991  042034  104005                  7$:     ERROR   5               ;JSR FAILED TO SAVE TH OLD PC
10992
10993  042036  010506                  6$:     MOV     R5,SP           ;RESTORE SP IN CASE ERROR SCREWED IT UP
10994
10995                                  ;;*****************************************************************
10996                                  ;*TEST 614      JSR MODE 1 TEST - N:C = 0000
10997                                  ;;*****************************************************************
10998  042040                          TST614:
10999  042040  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11000  042042  012700  000614                  MOV     #614,R0         ;;LOAD R0 WITH TEST NUMBER
11001  042046  013701  042102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11002                          .SBTTL USER CONTROLLED BREAKPOINT -- BIT12
11003  042052  032737  010000  063160          BIT     #BIT12,@#BPTLOC ;BREAKPOINT HALT SET ??
11004  042060  001401                          BEQ     .+4             ;BR IF NOT
11005  042062  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
11006  042064  010605                          MOV     SP,R5           ;SAVE THE SP
11007  042066  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11008  042072  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11009  042074  012702  042106                  MOV     #4$,R2          ;DEST ADDR = 4$
11010  042100  000257                          CCC                     ;N:C = 0000
11011
11012  042102  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11013
11014  042104  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11015
11016  042106  100403                  4$:     BMI     5$              ;N:C = 0000 ?
11017  042110  001402                          BEQ     5$
11018  042112  102401                          BVS     5$
11019  042114  103001                          BCC     6$
11020
11021  042116  104005                  5$:     ERROR   5               ;JSR FAILED - ALTERED FLAGS
11022
11023  042120  010506                  6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
```

```
11024                                   ;;*****************************************************************
11025                                   ;*TEST 615      JSR MODE 1 TEST - N:C = 1111
11026                                   ;;*****************************************************************
11027  042122                           TST615:
11028  042122  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11029  042124  012700  000615                  MOV     #615,R0         ;;LOAD R0 WITH TEST NUMBER
11030  042130  013701  042152                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11031  042134  010605                          MOV     SP,R5           ;SAVE THE SP
11032  042136  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11033  042142  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11034  042144  012702  042156                  MOV     #4$,R2          ;DEST ADDR = 4$
11035  042150  000277                          SCC                     ;N:C = 1111
11036
11037  042152  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11038
11039  042154  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11040
11041  042156  100003                  4$:     BPL     5$              ;N:C = 1111 ?
11042  042160  001002                          BNE     5$
11043  042162  102001                          BVC     5$
11044  042164  103401                          BCS     6$
11045  042166  104005                  5$:     ERROR   5               ;JSR ALTERED FLAGS
11046
11047  042170  010506                  6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11048
11049                                   ;;*****************************************************************
11050                                   ;*TEST 616      JSR MODE 2 TEST
11051                                   ;;*****************************************************************
11052  042172                           TST616:
11053  042172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11054  042174  012700  000616                  MOV     #616,R0         ;;LOAD R0 WITH TEST NUMBER
11055  042200  013701  042222                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11056  042204  010605                          MOV     SP,R5           ;SAVE THE SP
11057  042206  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11058  042212  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11059  042214  012702  042226                  MOV     #4$,R2          ;DEST ADDR = 4$
11060  042220  000257                          CCC                     ;SCOPE SYNC
11061
11062  042222  004422                  2$:     JSR     R4,(R2)+        ;TEST THE JSR - GO TO 4$
11063
11064  042224  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11065
11066  042226  005726                  4$:     TST     (SP)+           ;RESET SP
11067  042230  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11068  042232  001406                          BEQ     TST617          ;;BR IF YES
11069
11070  042234  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11071  042236  010603                          MOV     SP,R3           ;WAS SP
11072  042240  010504                          MOV     R5,R4
11073  042242  005744                          TST     -(R4)           ;S/B SP
11074  042244  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11075
11076  042246  010506                          MOV     R5,SP           ;RESTORE SP JUST IN CASE
11077
11078                                   ;;*****************************************************************
11079                                   ;*TEST 617      JSR MODE 3 TEST
```

```
11080                                   ;;*****************************************************************
11081  042250                           ;TST617:
11082  042250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11083  042252  012700  000617                  MOV     #617,R0         ;;LOAD R0 WITH TEST NUMBER
11084  042256  013701  042300                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11085  042262  010605                          MOV     SP,R5           ;SAVE THE SP
11086  042264  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11087  042270  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11088  042272  012702  042326                  MOV     #7$,R2          ;DEST ADDR = [7$]
11089  042276  000257                          CCC                     ;SCOPE SYNC
11090
11091  042300  004432                  2$:     JSR     R4,@(R2)+       ;TEST THE JSR - GO TO 4$ VIA 7$
11092
11093  042302  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11094
11095  042304  005726                  4$:     TST     (SP)+           ;RESET SP
11096  042306  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11097  042310  001411                          BEQ     TST620          ;;BR IF YES
11098
11099  042312  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11100  042314  010603                          MOV     SP,R3           ;WAS SP
11101  042316  010504                          MOV     R5,R4
11102  042320  005744                          TST     -(R4)           ;S/B SP
11103  042322  104003                  5$:     ERROR   3               ;JSR FAILED
11104  042324  000402                          BR      6$              ;GO EXIT
11105
11106  042326  042304                  7$:     4$                      ;CONTAINS JUMP ADDR
11107  042330  104006                          ERROR   6               ;JSR EXECUTED LIKE A MODE 1 OR 2
11108
11109  042332  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11110
11111                                   ;;*****************************************************************
11112                                   ;*TEST 620      JSR MODE 4 TEST
11113                                   ;;*****************************************************************
11114  042334                           ;TST620:
11115  042334  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11116  042336  012700  000620                  MOV     #620,R0         ;;LOAD R0 WITH TEST NUMBER
11117  042342  013701  042364                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11118  042346  010605                          MOV     SP,R5           ;SAVE THE SP
11119  042350  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11120  042354  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11121  042356  012702  042372                  MOV     #5$,R2          ;DEST ADDR = 4$+2
11122  042362  000257                          CCC                     ;SCOPE SYNC
11123
11124  042364  004442                  2$:     JSR     R4,-(R2)        ;TEST THE JSR - GO TO 4$
11125
11126  042366  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11127
11128  042370  000401                  4$:     BR      6$              ;JUMPED OK - GO CHECK SP
11129  042372  104005                  5$:     ERROR   5               ;JSR FAILED TO DECREMENT DEST REG
11130
11131  042374  005726                  6$:     TST     (SP)+           ;RESET SP
11132  042376  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11133  042400  001406                          BEQ     TST621          ;;BR IF YES
11134
11135  042402  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
```

```
11136  042404  010603                          MOV     SP,R3           ;WAS SP
11137  042406  010504                          MOV     R5,R4
11138  042410  005744                          TST     -(R4)           ;S/B SP
11139  042412  104003                  7$:     ERROR   3               ;JSR FAILED TO PUSH SP
11140
11141  042414  010506                  8$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11142
11143                                  ;;**********************************************************************
11144                                  ;*TEST 621       JSR MODE 5 TEST
11145                                  ;;**********************************************************************
11146  042416                          TST621:
11147  042416  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11148  042420  012700  000621                  MOV     #621,R0         ;;LOAD R0 WITH TEST NUMBER
11149  042424  013701  042446                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11150  042430  010605                          MOV     SP,R5           ;SAVE THE SP
11151  042432  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11152  042436  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11153  042440  012702  042476                  MOV     #7$,R2          ;DEST ADDR = [7$ - 2]
11154  042444  000257                          CCC                     ;SCOPE SYNC
11155
11156  042446  004452                  2$:     JSR     R4,@-(R2)       ;TEST THE JSR - GO TO 4$
11157
11158  042450  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11159
11160  042452  005726                  4$:     TST     (SP)+           ;RESET SP
11161  042454  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11162  042456  001411                          BEQ     TST622          ;;BR IF YES
11163
11164  042460  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11165  042462  010603                          MOV     SP,R3           ;WAS SP
11166  042464  010504                          MOV     R5,R4
11167  042466  005744                          TST     -(R4)           ;S/B SP
11168  042470  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11169  042472  000402                          BR      6$              ;GO EXIT
11170
11171  042474  042452                          4$                      ;CONTAINS JUMP ADDRESS
11172  042476  104005                  7$:     ERROR   5               ;JSR EXECUTED LIKE A MODE 1 OR 2
11173
11174  042500  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11175
11176                                  ;;**********************************************************************
11177                                  ;*TEST 622       JSR MODE 6 TEST
11178                                  ;;**********************************************************************
11179  042502                          TST622:
11180  042502  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11181  042504  012700  000622                  MOV     #622,R0         ;;LOAD R0 WITH TEST NUMBER
11182  042510  013701  042532                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11183  042514  010605                          MOV     SP,R5           ;SAVE THE SP
11184  042516  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11185  042522  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11186  042524  012702  042536                  MOV     #3$,R2          ;[R2] = BASE DEST ADDR
11187  042530  000257                          CCC                     ;SCOPE SYNC
11188
11189  042532  004462  000002          2$:     JSR     R4,4$-3$(R2)    ;TEST THE JSR - GO TO 4$
11190
11191  042536  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC OR INDEX FAILED
```

```
11192
11193  042540  005726                  4$:     TST     (SP)+           ;RESET SP
11194  042542  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11195  042544  001406                          BEQ     TST623          ;;BR IF YES
11196
11197  042546  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11198  042550  010603                          MOV     SP,R3           ;WAS SP
11199  042552  010504                          MOV     R5,R4
11200  042554  005744                          TST     -(R4)           ;S/B SP
11201  042556  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11202  042560  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
11203
11204                                  ;;**********************************************************************
11205                                  ;*TEST 623       JSR MODE 7 TEST
11206                                  ;;**********************************************************************
11207  042562                          TST623:
11208  042562  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11209  042564  012700  000623                  MOV     #623,R0         ;;LOAD R0 WITH TEST NUMBER
11210  042570  013701  042612                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11211  042574  010605                          MOV     SP,R5           ;SAVE THE SP
11212  042576  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11213  042602  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11214  042604  012702  042616                  MOV     #3$,R2          ;BASE DEST ADDR = 3$
11215  042610  000257                          CCC                     ;SCOPE SYNC
11216
11217  042612  004472  000024          2$:     JSR     R4,@7$-3$(R2)   ;TEST THE JSR - GO TO 4$ VIA 7$
11218
11219  042616  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11220                                                                  ;OR THE INDEX FAILED
11221
11222  042620  005726                  4$:     TST     (SP)+           ;RESET SP
11223  042622  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11224  042624  001411                          BEQ     TST624          ;;BR IF YES
11225
11226  042626  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11227  042630  010603                          MOV     SP,R3           ;WAS SP
11228  042632  010504                          MOV     R5,R4
11229  042634  005744                          TST     -(R4)           ;S/B SP
11230  042636  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH STACK
11231  042640  000402                          BR      6$              ;SKIP TO EXIT
11232
11233  042642  042620                  7$:     4$                      ;CONTAINS JUMP ADDR
11234  042644  104005                          ERROR   5               ;JSR WORKED LIKE A MODE 1 OR 2
11235
11236  042646  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11237
11238                                  ;;**********************************************************************
11239                                  ;*TEST 624       SOB TEST, [R] = 1, NO BRANCH
11240                                  ;;**********************************************************************
11241  042650                          TST624:
11242  042650  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11243  042652  012700  000624                  MOV     #624,R0         ;;LOAD R0 WITH TEST NUMBER
11244  042656  013701  042676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11245  042662  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
11246  042666  000402                          BR      2$-2            ;GO DO THE SOB
11247
```

```
11248  042670  104006                  3$:     ERROR   6               ;SOB SHOULDN'T HAVE BRANCHED HERE
11249  042672  000402                          BR      TST625          ;;GO TO SCOPE CALL
11250
11251  042674  000257                          CCC                     ;SYNC INSTR.
11252  042676  077204                  2$:     SOB     R2,3$           ;TEST THE SOB
11253
11254                                  ;;**********************************************************************
11255                                  ;*TEST 625      SOB TEST, [R] = 5, BRANCH 4 TIMES
11256                                  ;;**********************************************************************
11257  042700                          TST625:
11258  042700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11259  042702  012700  000625                  MOV     #625,R0         ;;LOAD R0 WITH TEST NUMBER
11260  042706  013701  042740                  MOV     @#SOB2,R1       ;GET COPY OF TEST INSTRUCTION WORD
11261  042712  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
11262  042716  012705  177773                  MOV     #-5,R5          ;SET UP R5 TO COUNT 5 BRANCHES
11263  042722  000405                          BR      SOB2-2          ;GO DO THE SOB
11264
11265  042724  000474                  SOB1:   BR      SOB3            ;USED BY LAST SOB TEST TO TEST MAX OFFSET
11266  042726  000240                          NOP                     ;OFFSET ADJUSTMENT
11267  042730  000240                          NOP
11268
11269  042732  005205                  SOB5:   INC     R5              ;COUNT ONE BRANCH
11270  042734  001406                          BEQ     SOBERR          ;BR IF TOO MANY LOOPS BY SOB
11271
11272  042736  000257                          CCC                     ;SCOPE SYNC
11273  042740  077204                  SOB2:   SOB     R2,SOB5         ;TEST THE SOB
11274  042742  005702                          TST     R2              ;R2 SHOULD CONTAIN 0
11275  042744  001403                          BEQ     TST626          ;;BR IF IT DOES
11276
11277  042746  104006                          ERROR   6               ;SOB COUNTER NOT ZERO
11278  042750  000401                          BR      TST626          ;;GO TO SCOPE CALL
11279  042752  104006                  SOBERR: ERROR   6               ;SOB MADE TOO MANY BRANCHES
11280
11281                                  ;;**********************************************************************
11282                                  ;*TEST 626      SOB TEST, [R] = 1, FLAGS = 1111
11283                                  ;;**********************************************************************
11284  042754                          TST626:
11285  042754  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11286  042756  012700  000626                  MOV     #626,R0         ;;LOAD R0 WITH TEST NUMBER
11287  042762  013701  042774                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11288  042766  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
11289  042772  000277                          SCC                     ;MAKE N:C = 1111
11290
11291  042774  077202                  2$:     SOB     R2,2$-2         ;TEST THE SOB
11292
11293  042776  103003                          BCC     3$              ;BR IF C = 0
11294  043000  102002                          BVC     3$              ;BR IF V = 0
11295  043002  001001                          BNE     3$              ;BR IF Z = 0
11296  043004  100401                          BMI     TST627          ;;BR IF N = 1
11297
11298  043006  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
11299
11300                                  ;;**********************************************************************
11301                                  ;*TEST 627      SOB TEST, [R] = 1, FLAGS = 0000
11302                                  ;;**********************************************************************
11303  043010                          TST627:
```

```
11304  043010  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11305  043012  012700  000627                  MOV     #627,R0         ;;LOAD R0 WITH TEST NUMBER
11306  043016  013701  043030                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11307  043022  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
11308  043026  000257                          CCC                     ;MAKE N:C = 0000
11309
11310  043030  077202                  2$:     SOB     R2,2$-2         ;TEST THE SOB
11311
11312  043032  103403                          BCS     3$              ;BR IF C = 1
11313  043034  102402                          BVS     3$              ;BR IF V = 1
11314  043036  001401                          BEQ     3$              ;BR IF Z = 1
11315  043040  100001                          BPL     TST630          ;;BR IF N = 0
11316
11317  043042  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
11318
11319                                  ;;**********************************************************************
11320                                  ;*TEST 630      SOB TEST, [R] = 5, FLAGS = 1111
11321                                  ;;**********************************************************************
11322  043044                          TST630:
11323  043044  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11324  043046  012700  000630                  MOV     #630,R0         ;;LOAD R0 WITH TEST NUMBER
11325  043052  013701  043064                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11326  043056  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
11327  043062  000277                          SCC                     ;MAKE N:C = 1111
11328
11329  043064  077201                  2$:     SOB     R2,2$           ;TEST THE SOB
11330
11331  043066  103003                          BCC     3$              ;BR IF C = 0
11332  043070  102002                          BVC     3$              ;BR IF V = 0
11333  043072  001001                          BNE     3$              ;BR IF Z = 0
11334  043074  100401                          BMI     TST631          ;;BR IF N = 1
11335
11336  043076  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
11337
11338                                  ;;**********************************************************************
11339                                  ;*TEST 631      SOB TEST, [R] = 5, FLAGS = 0000
11340                                  ;;**********************************************************************
11341  043100                          TST631:
11342  043100  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11343  043102  012700  000631                  MOV     #631,R0         ;;LOAD R0 WITH TEST NUMBER
11344  043106  013701  043120                  MOV     @#SOB4,R1       ;GET COPY OF TEST INSTRUCTION WORD
11345  043112  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
11346  043116  000257                  SOB3:   CCC                     ;MAKE N:C = 0000
11347
11348  043120  077277                  SOB4:   SOB     R2,SOB1         ;TEST THE SOB
11349
11350  043122  103403                          BCS     3$              ;BR IF C = 1
11351  043124  102402                          BVS     3$              ;BR IF V = 1
11352  043126  001401                          BEQ     3$              ;BR IF Z = 1
11353  043130  100001                          BPL     TST632          ;;BR IF N = 0
11354
11355  043132  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
11356
11357                                  ;;**********************************************************************
11358                                  ;*TEST 632      RTS TEST - N:C = 0000
11359                                  ;;**********************************************************************
```

```
11360  043134                          TST632:
11361  043134  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11362  043136  012700  000632                  MOV     #632,R0         ;;LOAD R0 WITH TEST NUMBER
11363  043142  013701  043174                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11364  043146  010605                          MOV     SP,R5           ;SAVE THE SP
11365  043150  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11366  043154  012704  177777          1$:     MOV     #-1,R4          ;R3 SHOULD GET 177777
11367  043160  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11368  043162  012703  043202                  MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM [R3]
11369  043166  012746  177777                  MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 WITH 177777
11370  043172  000257                          CCC                     ;N:C = 0000
11371
11372  043174  000203          2$:     RTS     R3                      ;TEST THE RTS - GO TO 4$
11373
11374  043176  104005          3$:     ERROR   5                       ;RTS FAILED TO LOAD THE PC
11375  043200  000415                  BR      10$                     ;GO TO EXIT - SCHOOLS OUT
11376
11377  043202  100403          4$:     BMI     5$                      ;N:C = 0000 ?
11378  043204  001402                  BEQ     5$
11379  043206  102401                  BVS     5$
11380  043210  103001                  BCC     6$
11381
11382  043212  104005          5$:     ERROR   5                       ;RTS ALTERED CODES - CLEARED ONE
11383
11384  043214  020403          6$:     CMP     R4,R3                   ;DID R3 GET LOADED FROM STACK ?
11385  043216  001401                  BEQ     8$                      ;BR IF YES
11386
11387  043220  104002          7$:     ERROR   2                       ;RTS FAILED TO LOAD REG
11388
11389  043222  020506          8$:     CMP     R5,SP                   ;DID RTS POP THE STACK POINTER ?
11390  043224  001404                  BEQ     TST633                  ;;BR IF YES
11391
11392  043226  010504                  MOV     R5,R4                   ;[R4] = S / B SP
11393  043230  010603                  MOV     SP,R3                   ;[R3] = WAS SP
11394  043232  104003          9$:     ERROR   3                       ;RTS FAILED TO POP SP
11395
11396  043234  010506          10$:    MOV     R5,SP                   ;FIX THE SP
11397
11398                                  ;;**************************************************************
11399                                  ;*TEST 633      RTT TEST - N:C = 1111
11400                                  ;;**************************************************************
11401  043236                          TST633:
11402  043236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11403  043240  012700  000633                  MOV     #633,R0         ;;LOAD R0 WITH TEST NUMBER
11404  043244  013701  043306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11405  043250  012702  177776                  MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
11406  043254  010605                          MOV     SP,R5           ;SAVE THE SP
11407  043256  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11408  043262  010506          1$:     MOV     R5,SP                   ;RESET SP FOR ERROR LOOP
11409  043264  012704  000340                  MOV     #340,R4         ;[R4] = S / B PSW AT HTIS POINT
11410  043270  012746  000340                  MOV     #340,-(SP)      ;NEW PSW S / B = 340
11411  043274  012746  043314                  MOV     #4$,-(SP)       ;NEW PC S / B = 4$
11412  043300  005037  177776                  CLR     @#PSW           ;CLEAR THE PSW
11413  043304  000277                          SCC                     ;N:C = 1111
11414
11415  043306  000006          2$:     RTT                             ;TEST THE RTT - GO TO 4$
```

```
11416
11417  043310  104005                  3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
11418  043312  000412                          BR      8$              ;GO TO EXIT - SCHOOL'S OUT
11419
11420  043314  013703  177776          4$:     MOV     @#PSW,R3        ;SAVE THE PSW
11421  043320  020403                          CMP     R4,R3           ;WAS PSW = 340 ?
11422  043322  001401                          BEQ     6$              ;BR IF IT WAS
11423
11424  043324  104001                  5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
11425
11426  043326  020506                  6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
11427  043330  001404                          BEQ     TST634          ;;BR IF YES
11428
11429  043332  010504                          MOV     R5,R4           ;[R4] = S / B SP
11430  043334  010603                          MOV     SP,R3           ;[R3] = WAS SP
11431  043336  104003                  7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
11432
11433  043340  010506                  8$:     MOV     R5,SP           ;FIX THE SP
11434
11435                                   ;;*********************************************************************
11436                                   ;*TEST 634      RTT TEST - N:C = 0000
11437                                   ;;*********************************************************************
11438  043342                          TST634:
11439  043342  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11440  043344  012700  000634                  MOV     #634,R0         ;;LOAD R0 WITH TEST NUMBER
11441  043350  013701  043414                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11442  043354  012702  177776                  MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
11443  043360  010605                          MOV     SP,R5           ;SAVE THE SP
11444  043362  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11445  043366  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11446  043370  012704  000017                  MOV     #017,R4         ;[R4] = S / B PSW AT HTIS POINT
11447  043374  012746  000017                  MOV     #017,-(SP)      ;NEW PSW S / B = 017
11448  043400  012746  043422                  MOV     #4$,-(SP)       ;NEW PC S / B = 4$
11449  043404  012737  000340  177776          MOV     #340,@#PSW      ;MAKE [PSW] = 340
11450  043412  000257                          CCC                     ;N:C = 0000
11451
11452  043414  000006                  2$:     RTT                     ;TEST THE RTT - GO TO 4$
11453
11454  043416  104005                  3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
11455  043420  000412                          BR      8$              ;GO TO EXIT - SCHOOL'S OUT
11456
11457  043422  013703  177776          4$:     MOV     @#PSW,R3        ;SAVE THE PSW
11458  043426  020403                          CMP     R4,R3           ;WAS PSW = 017 ?
11459  043430  001401                          BEQ     6$              ;BR IF IT WAS
11460
11461  043432  104001                  5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
11462
11463  043434  020506                  6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
11464  043436  001404                          BEQ     TST635          ;;BR IF YES
11465
11466  043440  010504                          MOV     R5,R4           ;[R4] = S / B SP
11467  043442  010603                          MOV     SP,R3           ;[R3] = WAS SP
11468  043444  104003                  7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
11469
11470  043446  010506                  8$:     MOV     R5,SP           ;FIX THE SP
11471
```

```
11472                                   ;;*****************************************************************
11473                                   ;*TEST 635      MARK INSTRUCTION TEST - N:C=0000
11474                                   ;;*****************************************************************
11475  043450                           TST635:
11476  043450  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11477  043452  012700  000635                  MOV     #635,R0         ;;LOAD R0 WITH TEST NUMBER
11478  043456  013701  043502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11479  043462  010602                          MOV     SP,R2           ;SAVE SP
11480  043464  012704  125252                  MOV     #125252,R4      ;[R5] SHOULD BE 125252
11481  043470  012705  043532                  MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
11482  043474  010437  043516                  MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
11483  043500  000257                          CCC                     ;N:C=0000
11484
11485  043502  006405                  2$:     MARK+5                  ;TEST THE MARK
11486
11487  043504  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
11488  043510  010206                          MOV     R2,SP           ;RESET SP
11489  043512  104005                  3$:     ERROR   5               ;MARK FAILED TO EXECUTE
11490
11491  043514  000444                          BR      TST636          ;;GO TO SCOPE EXIT
11492
11493  043516  125252                  6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
11494
11495  043520  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
11496  043524  010206                          MOV     R2,SP           ;RESET SP
11497  043526  104005                  5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
11498
11499  043530  000436                          BR      TST636          ;;GO TO SCOPE EXIT
11500
11501  043532  100403                  4$:     BMI     10$             ;N:C=0000?
11502  043534  001402                          BEQ     10$
11503  043536  102401                          BVS     10$
11504  043540  103011                          BCC     8$
11505
11506  043542  013703  177776          10$:    MOV     @#PSW,R3        ;SAVE FLAGS IN R3
11507  043546  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
11508  043552  010206                          MOV     R2,SP           ;RESET SP
11509  043554  012[illegible]  177776          MOV     #PSW,R2         ;DEST=PSW
11510  043560  104007                  7$:     ERROR   7               ;MARK SET A FLAG
11511  043562  000421                          BR      TST636          ;;GO TO SCOPE EXIT
11512
11513  043564  020627  043520          8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
11514  043570  001406                          BEQ     11$             ;BR IF YES
11515  043572  010603                          MOV     SP,R3           ;PUT BAD SP IN R3
11516  043574  012704  043520                  MOV     #6$+2,R4        ;S/B SP
11517  043600  010206                          MOV     R2,SP           ;RESET SP
11518  043602  104003                  9$:     ERROR   3               ;MARK FAILED TO RESET SP
11519
11520  043604  000410                          BR      TST636          ;;GO TO SCOPE EXIT
11521
11522  043606  020504                  11$:    CMP     R5,R4           ;DID MARK RESTORE OLD R5
11523  043610  001405                          BEQ     12$             ;BR IF YES
11524
11525  043612  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
11526  043616  010503                          MOV     R5,R3           ;WAS DEST
11527  043620  010206                          MOV     R2,SP           ;RESET SP
```

```
11528  043622  104004                    ERROR   4               ;MARK FAILED TO RESET R5
11529
11530  043624  010206          12$:      MOV     R2,SP           ;RESET SP
11531
11532                          ;;*****************************************************************
11533                          ;*TEST 636      MARK INSTRUCTION TEST - N:C=1111
11534                          ;;*****************************************************************
11535  043626                  TST636:
11536  043626  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11537  043630  012700  000636            MOV     #636,R0         ;;LOAD R0 WITH TEST NUMBER
11538  043634  013701  043660            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11539  043640  010602                    MOV     SP,R2           ;SAVE SP
11540  043642  012704  125252            MOV     #125252,R4      ;[R5] SHOULD BE 125252
11541  043646  012705  043710            MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
11542  043652  010437  043674            MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
11543  043656  000277                    SCC                     ;N:C=1111
11544
11545  043660  006405          2$:       MARK+5                  ;TEST THE MARK
11546
11547  043662  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
11548  043666  010206                    MOV     R2,SP           ;RESET SP
11549  043670  104005          3$:       ERROR   5               ;MARK FAILED TO EXECUTE
11550
11551  043672  000444                    BR      TST637          ;;GO TO SCOPE EXIT
11552
11553  043674  125252          6$:       125252                  ;THIS WORD SHOULD GET LOADED INTO R5
11554
11555  043676  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
11556  043702  010206                    MOV     R2,SP           ;RESET SP
11557  043704  104005          5$:       ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
11558
11559  043706  000436                    BR      TST637          ;;GO TO SCOPE EXIT
11560
11561  043710  100003          4$:       BPL     7$              ;N:C=1111
11562  043712  001002                    BNE     7$
11563  043714  102001                    BVC     7$
11564  043716  103411                    BCS     8$
11565
11566  043720  013703  177776    7$:     MOV     @#PSW,R3        ;SAVE FLAGS IN R3
11567  043724  010637  001074            MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
11568  043730  010206                    MOV     R2,SP           ;RESET SP
11569  043732  012702  177776            MOV     #PSW,R2         ;DEST=PSW
11570  043736  104007                    ERROR   7               ;MARK SET A FLAG
11571  043740  000421                    BR      TST637          ;;GO TO SCOPE EXIT
11572
11573  043742  020627  043676    8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
11574  043746  001406                    BEQ     9$              ;BR IF YES
11575  043750  010603                    MOV     SP,R3           ;PUT BAD SP IN R3
11576  043752  012704  043676            MOV     #6$+2,R4        ;S/B SP
11577  043756  010206                    MOV     R2,SP           ;RESET SP
11578  043760  104003                    ERROR   3               ;MARK FAILED TO RESET SP
11579
11580  043762  000410                    BR      TST637          ;;GO TO SCOPE EXIT
11581
11582  043764  020504    9$:     CMP     R5,R4           ;DID MARK RESTORE OLD R5
11583  043766  001405            BEQ     10$             ;BR IF YES
```

```
11584
11585  043770  010637  001074                MOV     SP,@#SREG5      ;SAVE BAD SP FOR PRINTING
11586  043774  010503                        MOV     R5,R3           ;WAS DEST
11587  043776  010206                        MOV     R2,SP           ;RESET SP
11588  044000  104004                        ERROR   4               ;MARK FAILED TO RESET R5
11589
11590  044002  010206              10$:      MOV     R2,SP           ;RESET SP
11591
11592                               ;;*****************************************************************
11593                               ;*TEST 637      BASIC LINE CLOCK RESPONSE TEST
11594                               ;;*****************************************************************
11595  044004                       TST637:
11596  044004  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11597  044006  012700  000637                MOV     #637,R0         ;;LOAD R0 WITH TEST NUMBER
11598  044012  013701  044042                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11599  044016  010605                        MOV     SP,R5           ;SAVE SP
11600  044020  012702  177546                MOV     #LKCSR,R2       ;(R2) = LINE CLOCK ADDRESS
11601  044024  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11602  044030  010506              1$:       MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11603  044032  012737  044046  000004        MOV     #4$,@#4         ;GO TO 4$ IF BUS TIMEOUT
11604  044040  000257                        CCC                     ;SCOPE SYNC
11605
11606  044042  005712              2$:       TST     (R2)            ;REFERENCE LKCSR ADDR
11607
11608  044044  000404                        BR      6$              ;GO TO EXIT
11609
11610  044046  012737  061144  000004  4$:   MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
11611  044054  104006              3$:       ERROR   6               ;LKCSR FAILED TO RESPOND
11612
11613  044056  010506              6$:       MOV     R5,SP           ;RESET SP
11614  044060  012737  061144  000004        MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
11615
11616                               ;;*****************************************************************
11617                               ;*TEST 640      LINE CLOCK TEST - LKCSR BIT 7 SET
11618                               ;;*****************************************************************
11619  044066                       TST640:
11620  044066  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11621  044070  012700  000640                MOV     #640,R0         ;;LOAD R0 WITH TEST NUMBER
11622  044074  013701  044112                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11623  044100  012702  177546                MOV     #LKCSR,R2       ;DEST ADDR = 177546
11624  044104  012704  000200                MOV     #200,R4         ;[LKCSR] S / B = 200
11625  044110  000257                        CCC                     ;SCOPE SYNC
11626
11627  044112  030412              2$:       BIT     R4,(R2)         ;TEST BIT 7 IN LKCSR
11628
11629  044114  001002                        BNE     TST641          ;;BR IF IT'S SET
11630
11631  044116  011203                        MOV     (R2),R3         ;GET WAS DATA
11632  044120  104001              3$:       ERROR   1               ;BIT 7 NOT SET IN LKCSR
11633
11634                               ;;*****************************************************************
11635                               ;*TEST 641      LINE CLOCK TEST - LKCSR BIT 6 CLEAR
11636                               ;;*****************************************************************
11637  044122                       TST641:
11638  044122  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11639  044124  012700  000641                MOV     #641,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
11640  044130  013701  044146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11641  044134  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
11642  044140  012704  000200                  MOV     #200,R4         ;[LKCSR] S / B = 200
11643  044144  000257                          CCC                     ;SCOPE SYNC
11644
11645  044146  032712  000100          2$:     BIT     #100,(R2)       ;TEST BIT 6 IN LKCSR
11646
11647  044152  001402                          BEQ     TST642          ;;BR IF CLEAR
11648
11649  044154  011203                          MOV     (R2),R3         ;GET WAS DATA
11650  044156  104001          3$:     ERROR   1               ;BIT 6 (INTR. ENAB.) IN LKCSR WAS SET
11651
11652                          ;;*****************************************************************
11653                          ;*TEST 642      LINE CLOCK TEST - LKCSR BIT 6 SET
11654                          ;;*****************************************************************
11655  044160                  TST642:
11656  044160  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11657  044162  012700  000642                  MOV     #642,R0         ;;LOAD R0 WITH TEST NUMBER
11658  044166  013701  044236                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11659  044172  010605                          MOV     SP,R5           ;SAVE SP
11660  044174  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
11661  044200  012704  000300                  MOV     #300,R4         ;[LKCSR] S / B = 300
11662  044204  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11663  044210  012737  044252  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR IN CASE LOGIC
11664  044216  012737  000340  000102          MOV     #340,@#102      ;FAULT CAUSES ATL INTERRUPT
11665  044224  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11666  044226  012737  000340  177776          MOV     #340,@#PSW      ;SET PRIORITY TO LEVEL 7
11667  044234  000257                          CCC                     ;SCOPE SYNC
11668
11669  044236  052712  000100          2$:     BIS     #100,(R2)       ;SET BIT 6 IN LKCSR
11670
11671  044242  020412                          CMP     R4,(R2)         ;RESULT CORRECT?
11672  044244  001402                          BEQ     4$              ;BR IF YES
11673
11674  044246  011203                          MOV     (R2),R3         ;GET WAS DATA
11675  044250  104001          3$:     ERROR   1               ;BIT 6 FAILED TO SET IN LKCSR
11676
11677  044252  042737  000102  000100  4$:     BIC     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLOCK VECTOR
11678  044260  005037  000102                  CLR     @#102
11679  044264  042712  000100                  BIC     #100,(R2)       ;TURN OF LINE CLK INTR. ENAB.
11680  044270  010506                          MOV     R5,SP           ;RESET SP
11681
11682                          ;;*****************************************************************
11683                          ;*TEST 643      LINE CLK BASIC INTERRUPT TEST
11684                          ;;*****************************************************************
11685  044272                  TST643:
11686  044272  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11687  044274  012700  000643                  MOV     #643,R0         ;;LOAD R0 WITH TEST NUMBER
11688  044300  013701  044346                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11689  044304  010605                          MOV     SP,R5           ;SAVE SP
11690  044306  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
11691  044312  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11692  044316  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11693  044320  005004                          CLR     R4              ;INITIALIZE TIMER
11694  044322  012737  044364  000100          MOV     #4$,@#100       ;SET UP LINE CLOCK VECTOR TO TO
11695  044330  012737  000340  000102          MOV     #340,@#102      ;TO 4$ WITH PROCESSOR PRIORITY = 7
```

```
11696  044336  005012                          CLR     (R2)            ;CLEAR LKCSR
11697  044340  005037  177776                  CLR     @#PSW           ;SET PRIORITY TO LEVEL 000
11698  044344  000257                          CCC                     ;SCOPE SYNC
11699
11700  044346  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPT
11701
11702  044352  005304                          DEC     R4              ;WAIT FOR INTR - REPORT ERROR IF
11703  044354  001376                          BNE     .-2             ;R4 GOES TO 000000
11704
11705  044356  042712  000100                  BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
11706  044362  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTERRUPT
11707
11708  044364  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
11709  044370  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLK VECTOR
11710  044376  005037  000102                  CLR     @#102
11711  044402  010506                          MOV     R5,SP           ;RESET SP
11712  044404  005037  177776                  CLR     @#PSW           ;RESET PRIORITY TO LEVEL 0
11713
11714                                  ;;**********************************************************************
11715                                  ;*TEST 644       RESET TEST - N:C = 1111
11716                                  ;;**********************************************************************
11717  044410                          TST644:
11718  044410  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11719  044412  012700  000644                  MOV     #644,R0         ;;LOAD R0 WITH TEST NUMBER
11720  044416  013701  044450                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11721  044422  012737  000001  001110          MOV     #1,@#STIMES     ;NO ITERATIONS ON THIS TEST
11722  044430  012702  177564                  MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
11723  044434  012737  000340  177776          MOV     #340,@#PSW      ;MAKE PRTY. BITS ALL 1'S
11724  044442  052712  000004                  BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
11725  044446  000277                          SCC                     ;N:C = 1111
11726
11727  044450  000005                  2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
11728
11729  044452  013705  177776                  MOV     @#PSW,R5        ;SAVE THE PSW
11730  044456  032712  000004                  BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
11731  044462  001403                          BEQ     4$              ;BR IF YES
11732
11733  044464  042712  000004                  BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
11734  044470  104006                  3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
11735
11736  044472  022705  000357          4$:     CMP     #357,R5         ;DID RESET ALTER THE PSW ??
11737  044476  001406                          BEQ     6$              ;BR IF NOT
11738
11739  044500  012704  000357                  MOV     #357,R4         ;[R4] = S/B PSW
11740  044504  010503                          MOV     R5,R3           ;[R3] = WAS PSW
11741  044506  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
11742  044512  104001                  5$:     ERROR   1               ;RESET ALTERED THE PSW
11743
11744  044514  005037  177776          6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
11745  044520  042737  000004  177564          BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
11746
11747                                  ;;**********************************************************************
11748                                  ;*TEST 645       RESET TEST - N:C = 0000
11749                                  ;;**********************************************************************
11750  044526                          TST645:
11751  044526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
11752  044530  012700  000645          MOV     #645,R0         ;;LOAD R0 WITH TEST NUMBER
11753  044534  013701  044564          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11754  044540  012737  000001  001110  MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
11755  044546  012702  177564          MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
11756  044552  005037  177776          CLR     @#PSW           ;MAKE PRTY. BITS ALL 0'S
11757  044556  052712  000004          BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
11758  044562  000257                  CCC                     ;N:C = 0000
11759
11760  044564  000005          2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
11761
11762  044566  013705  177776          MOV     @#PSW,R5        ;SAVE THE PSW
11763  044572  032712  000004          BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
11764  044576  001403                  BEQ     4$              ;BR IF YES
11765
11766  044600  042712  000004          BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
11767  044604  104006          3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
11768
11769  044606  022705  000000  4$:     CMP     #0,R5           ;DID RESET ALTER THE PSW ??
11770  044612  001406                  BEQ     6$              ;BR IF NOT
11771
11772  044614  012704  000357          MOV     #357,R4         ;[R4] = S/B PSW
11773  044620  010503                  MOV     R5,R3           ;[R3] = WAS PSW
11774  044622  012702  177776          MOV     #PSW,R2         ;DEST = PSW
11775  044626  104001          5$:     ERROR   1               ;RESET ALTERED THE PSW
11776
11777  044630  005037  177776  6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
11778  044634  042737  000004  177564  BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
11779
11780                          ;;*****************************************************************
11781                          ;*TEST 646      WAIT INSTRUCTION TEST - [PSW] = 151
11782                          ;;*****************************************************************
11783  044642                  ;ST646:
11784  044642  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11785  044644  012700  000646          MOV     #646,R0         ;;LOAD R0 WITH TEST NUMBER
11786  044650  013701  044736          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11787  044654  010605                  MOV     SP,R5           ;SAVE THE SP
11788  044656  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11789  044662  012702  177564  1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
11790  044666  012737  044754  000064  MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
11791  044674  012737  000200  000066  MOV     #200,@#66       ;AT LEVEL 4
11792  044702  010506                  MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11793  044704  005012                  CLR     (R2)            ;INIT DL11 XCSR
11794  044706  005003                  CLR     R3              ;INIT TIMER
11795
11796  044710  105712          3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
11797  044712  100403                  BMI     5$              ;BR IF YES
11798  044714  005303                  DEC     R3              ;COUNT THE TIMER
11799  044716  001374                  BNE     3$              ;BR IF NO TIMEOUT
11800  044720  000440                  BR      9$              ;GO REPORT TIMEOUT
11801
11802  044722  012737  000140  177776  5$:     MOV     #140,@#PSW      ;SET PSW PRTY BITS TO LEVEL 3
11803  044730  000277                  SCC                     ;N:C=1111
11804  044732  152712  000100          BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1001
11805
11806  044736  000001          2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
11807
```

```
11808  044740  012737  000340  177776        MOV     #340,@#PSW      ;LOCK OUT INTR
11809  044746  005012                        CLR     (R2)            ;TURN OFF DL11 INTR ENAB
11810  044750  104006                        ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
11811  044752  000424                        BR      8$              ;GO EXIT THIS TEST
11812
11813  044754  042712  000100          4$:   BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
11814  044760  022716  044740                CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
11815  044764  001402                        BEQ     6$              ;BR IF YES
11816
11817  044766  104006                        ERROR   6               ;WAIT NOT FETCHED PROPERLY
11818  044770  000415                        BR      8$              ;GO EXIT THE TEST
11819
11820  044772  022766  000151  000002  6$:   CMP     #151,2(SP)      ;DID "WAIT" ALTER THE PSW ??
11821  045000  001411                        BEQ     8$              ;BR IF YES
11822
11823  045002  012704  000151                MOV     #151,R4         ;[R4] = S/B PSW
11824  045006  016603  000002                MOV     2(SP),R3        ;[R3] = WAS PSW
11825  045012  012702  177776                MOV     #PSW,R2         ;DEST = PSW
11826  045016  104001                  7$:   ERROR   1               ;"WAIT" ALTERED THE PSW
11827  045020  000401                        BR      8$              ;GOT TO EXIT TEST
11828
11829  045022  104006                  9$:   ERROR   6               ;DL11 FAILED TO SET READY ON TIME
11830
11831  045024  010506                  8$:   MOV     R5,SP           ;RESET THE SP
11832  045026  005037  177776                CLR     @#PSW           ;CLEAR OUT THE PSW
11833  045032  005012                        CLR     (R2)            ;TURN OFF DL11 INTR.
11834  045034  012737  000066  000064        MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
11835  045042  005037  000066                CLR     @#66
11836
11837                                  ;;*****************************************************************
11838                                  ;*TEST 647      WAIT INSTRUCTION TEST - [PSW] = 010
11839                                  ;;*****************************************************************
11840  045046                          TST647:
11841  045046  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11842  045050  012700  000647                MOV     #647,R0         ;;LOAD R0 WITH TEST NUMBER
11843  045054  013701  045140                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11844  045060  010605                        MOV     SP,R5           ;SAVE THE SP
11845  045062  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11846  045066  012702  177564          1$:   MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
11847  045072  012737  045156  000064        MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
11848  045100  012737  000200  000066        MOV     #200,@#66       ;AT LEVEL 4
11849  045106  010506                        MOV     R5,SP           ;RESET SP FOR ERROR LOOP
11850  045110  005012                        CLR     (R2)            ;INIT DL11 XCSR
11851  045112  005003                        CLR     R3              ;INIT TIMER
11852
11853  045114  105712                  3$:   TSTB    (R2)            ;DL11 XMIT READY SET ??
11854  045116  100403                        BMI     5$              ;BR IF YES
11855  045120  005303                        DEC     R3              ;COUNT THE TIMER
11856  045122  001374                        BNE     3$              ;BR IF NO TIMEOUT
11857  045124  000437                        BR      9$              ;GO REPORT TIMEOUT
11858
11859  045126  005037  177776          5$:   CLR     @#PSW           ;SET PSW PRTY BITS TO LEVEL 0
11860  045132  000257                        CCC                     ;N:C=0000
11861  045134  152712  000100                BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1000
11862
11863  045140  000001                  2$:   WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
```

```
11864
11865  045142  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
11866  045150  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
11867  045152  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
11868  045154  000424                          BR      8$              ;GO EXIT THIS TEST
11869
11870  045156  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
11871  045162  022716  045142                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
11872  045166  001402                          BEQ     6$              ;BR IF YES
11873
11874  045170  104006                          ERROR   -               ;WAIT NOT FETCHED PROPERLY
11875  045172  000415                          BR      8$              ;GO EXIT THE TEST
11876
11877  045174  022766  000010  000002  6$:     CMP     #010,2(SP)      ;DID "WAIT" ALTER THE PSW ??
11878  045202  001411                          BEQ     8$              ;BR IF NO
11879
11880  045204  012704  000010                  MOV     #010,R4         ;[R4] = S/B PSW
11881  045210  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
11882  045214  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
11883  045220  104001                  7$:     ERROR   1               ;"WAIT" ALTERED THE PSW
11884  045222  000401                          BR      8$              ;GOT TO EXIT TEST
11885
11886  045224  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
11887
11888  045226  010506                  8$:     MOV     R5,SP           ;RESET THE SP
11889  045230  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
11890  045234  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
11891  045236  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
11892  045244  005037  000066                  CLR     @#66
11893
11894                                  ;;*****************************************************************
11895                                  ;*TEST 650      BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
11896                                  ;;*****************************************************************
11897  045250                          TST650:
11898  045250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11899  045252  012700  000650                  MOV     #650,R0         ;;LOAD R0 WITH TEST NUMBER
11900  045256  013701  045324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11901  045262  010605                          MOV     SP,R5           ;SAVE THE SP
11902  045264  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11903  045270  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
11904  045274  012737  045342  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
11905  045302  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
11906  045310  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
11907  045312  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
11908  045314  012737  000040  177776          MOV     #40,@#PSW       ;SET CPU PRIORITY TO LEVEL 1
11909  045322  000257                          CCC                     ;SCOPE SYNC
11910
11911  045324  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
11912
11913  045330  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
11914  045332  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
11915
11916  045334  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
11917  045340  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 1
11918
11919  045342  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
```

```
11920  045346  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
11921  045354  005037  000102                  CLR     @#102
11922  045360  010506                          MOV     R5,SP           ;RESET THE SP
11923  045362  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
11924
11925                                  ;;*****************************************************************
11926                                  ;*TEST 651      BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
11927                                  ;;*****************************************************************
11928  045366                          ;ST651:
11929  045366  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11930  045370  012700  000651                  MOV     #651,R0         ;;LOAD R0 WITH TEST NUMBER
11931  045374  013701  045442                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11932  045400  010605                          MOV     SP,R5           ;SAVE THE SP
11933  045402  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11934  045406  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
11935  045412  012737  045460  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
11936  045420  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
11937  045426  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
11938  045430  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
11939  045432  012737  000100  177776          MOV     #100,@#PSW      ;SET CPU PRIORITY TO LEVEL 2
11940  045440  000257                          CCC                     ;SCOPE SYNC
11941
11942  045442  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
11943
11944  045446  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
11945  045450  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
11946
11947  045452  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
11948  045456  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 2
11949
11950  045460  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
11951  045464  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
11952  045472  005037  000102                  CLR     @#102
11953  045476  010506                          MOV     R5,SP           ;RESET THE SP
11954  045500  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
11955
11956                                  ;;*****************************************************************
11957                                  ;*TEST 652      BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
11958                                  ;;*****************************************************************
11959  045504                          ;ST652:
11960  045504  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11961  045506  012700  000652                  MOV     #652,R0         ;;LOAD R0 WITH TEST NUMBER
11962  045512  013701  045560                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11963  045516  010605                          MOV     SP,R5           ;SAVE THE SP
11964  045520  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11965  045524  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
11966  045530  012737  045576  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
11967  045536  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
11968  045544  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
11969  045546  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
11970  045550  012737  000140  177776          MOV     #140,@#PSW      ;SET CPU PRIORITY TO LEVEL 3
11971  045556  000257                          CCC                     ;SCOPE SYNC
11972
11973  045560  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
11974
11975  045564  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
```

```
11976  045566  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
11977
11978  045570  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
11979  045574  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 3
11980
11981  045576  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
11982  045602  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
11983  045610  005037  000102                  CLR     @#102
11984  045614  010506                          MOV     R5,SP           ;RESET THE SP
11985  045616  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
11986
11987                                  ;;*****************************************************************
11988                                  ;*TEST 653      BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
11989                                  ;;*****************************************************************
11990  045622                          TST653:
11991  045622  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11992  045624  012700  000653                  MOV     #653,R0         ;;LOAD R0 WITH TEST NUMBER
11993  045630  013701  045676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11994  045634  010605                          MOV     SP,R5           ;SAVE THE SP
11995  045636  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11996  045642  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
11997  045646  012737  045714  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
11998  045654  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
11999  045662  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12000  045664  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12001  045666  012737  000200  177776          MOV     #200,@#PSW      ;SET CPU PRIORITY TO LEVEL 4
12002  045674  000257                          CCC                     ;SCOPE SYNC
12003
12004  045676  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12005
12006  045702  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12007  045704  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12008
12009  045706  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12010  045712  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 4
12011
12012  045714  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12013  045720  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12014  045726  005037  000102                  CLR     @#102
12015  045732  010506                          MOV     R5,SP           ;RESET THE SP
12016  045734  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12017
12018                                  ;;*****************************************************************
12019                                  ;*TEST 654      BR PRIORITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12020                                  ;;*****************************************************************
12021  045740                          TST654:
12022  045740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12023  045742  012700  000654                  MOV     #654,R0         ;;LOAD R0 WITH TEST NUMBER
12024  045746  013701  046014                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12025  045752  010605                          MOV     SP,R5           ;SAVE THE SP
12026  045754  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12027  045760  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12028  045764  012737  046032  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12029  045772  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12030  046000  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12031  046002  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
```

```
12032  046004  012737  000240  177776          MOV     #240,@#PSW      ;SET CPU PRIORITY TO LEVEL 5
12033  046012  000257                          CCC                     ;SCOPE SYNC
12034
12035  046014  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12036
12037  046020  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12038  046022  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12039
12040  046024  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12041  046030  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 5
12042
12043  046032  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12044  046036  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12045  046044  005037  000102                  CLR     @#102
12046  046050  010506                          MOV     R5,SP           ;RESET THE SP
12047  046052  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12048
12049                          ;;*****************************************************************
12050                          ;*TEST 655      BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12051                          ;;*****************************************************************
12052  046056                  TST655:
12053  046056  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
12054  046060  012700  000655          MOV     #655,R0                 ;;LOAD R0 WITH TEST NUMBER
12055  046064  013701  046144          MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
12056                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT13
12057  046070  032737  020000  063160          BIT     #BIT13,@#BPTLOC ;BREAKPOINT HALT SET ??
12058  046076  001401                          BEQ     .+4             ;BR IF NOT
12059  046100  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
12060  046102  010605                          MOV     SP,R5           ;SAVE THE SP
12061  046104  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12062  046110  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12063  046114  012737  046156  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12064  046122  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12065  046130  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12066  046132  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12067  046134  012737  000300  177776          MOV     #300,@#PSW      ;SET CPU PRIORITY TO LEVEL 6
12068  046142  000257                          CCC                     ;SCOPE SYNC
12069
12070  046144  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12071
12072  046150  005304                          DEC     R4              ;COUNT UNTIL (R4) = 000000 - THEN
12073  046152  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12074  046154  000403                          BR      6$              ;GO TO EXIT - ALL OK
12075
12076  046156  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12077  046162  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 6
12078
12079  046164  042712  000100          6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12080  046170  012737  000102  000100          MOV     #102,@#100      ;RESET THE TRAP CATCHER IN THE VECTOR
12081  046176  005037  000102                  CLR     @#102
12082  046202  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12083  046204  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12084
12085                          ;;*****************************************************************
12086                          ;*TEST 656      BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12087                          ;;*****************************************************************
```

```
12088  046210                                TST656:
12089  046210  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12090  046212  012700  000656                MOV     #656,R0         ;;LOAD R0 WITH TEST NUMBER
12091  046216  013701  046264                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12092  046222  010605                        MOV     SP,R5           ;SAVE THE SP
12093  046224  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12094  046230  012702  177564        1$:     MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12095  046234  012737  046276  000064        MOV     #4$,@#64        ;IF INTR OCCURS - GO TO 4$
12096  046242  012737  000340  000066        MOV     #340,@#66       ;WITH CPU PRIORITY AT LEVEL 7
12097  046250  010506                        MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12098  046252  005004                        CLR     R4              ;INITIALIZE R4 AS TIMER
12099  046254  012737  000340  177776        MOV     #340,@#PSW      ;SET CPU PRIORITY TO LEVEL 7
12100  046262  000257                        CCC                     ;SCOPE SYNC
12101
12102  046264  052712  000100        2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12103
12104  046270  005304                        DEC     R4              ;COUNT UNTIL (R4) = 000000 - THEN
12105  046272  001376                        BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12106  046274  000403                        BR      6$              ;GO TO EXIT - ALL OK
12107
12108  046276  042712  000100        4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12109  046302  104006                3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 7
12110
12111  046304  042712  000100        6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12112  046310  012737  000066  000064        MOV     #66,@#64        ;RESET THE TRAP CATCHER IN THE VECTOR
12113  046316  005037  000066                CLR     @#66
12114  046322  010506                        MOV     R5,SP           ;RESET SP JUST IN CASE
12115  046324  005037  177776                CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12116
12117                                ;;*****************************************************************
12118                                ;*TEST 657      "CLR  @#PSW" ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12119                                ;THIS TEST VERIFIES THAT IF A "BR" REQUEST IS PENDING WHEN A "CLR  @#PSW"
12120                                ;IS EXECUTED TO LOWER THE CPU PRIORITY, THE REQUEST IS GRANTED BEFORE
12121                                ;EXECUTION OF THE INSTRUCTION FOLLOWING THE "CLR"
12122                                ;;*****************************************************************
12123  046330                        TST657:
12124  046330  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12125  046332  012700  000657                MOV     #657,R0         ;;LOAD R0 WITH TEST NUMBER
12126  046336  013701  046430                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12127  046342  012702  177546                MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12128  046346  010605                        MOV     SP,R5           ;SAVE THE SP
12129  046350  010737  001010                MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12130  046354  012737  046436  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR TO GO TO 4$
12131  046362  012737  000300  000102        MOV     #300,@#102
12132  046370  010506                        MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12133  046372  005004                        CLR     R4              ;INITIALIZE TIMER FO KW
12134  046374  005003                        CLR     R3              ;CLEAR SOFTWARE FLAG
12135  046376  012737  000340  177776        MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12136  046404  052712  000100                BIS     #100,(R2)       ;ENABLE LCLK INTRS
12137  046410  042712  000200                BIC     #200,(R2)       ;CLEAR LINE CLOCK READY
12138  046414  105712                11$:    TSTB    (R2)            ;LCLK READY TO INTR ??
12139  046416  100403                        BMI     12$             ;BR IF YES
12140  046420  005304                        DEC     R4              ;COUNT THE TIMER
12141  046422  001374                        BNE     11$             ;BR IF NO TIMEOUT
12142  046424  000411                        BR      6$              ;GO REPORT TIMEOUT
12143  046426  000257                12$:    CCC                     ;SCOPE SYNC
```

```
12144
12145  046430  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - LCLK SHOULD INTERRUPT
12146                                                                   ;BEFORE FETCHING NEXT INSTRUCTION
12147  046434  005103                          COM     R3              ;SHOULD NOT BE FETCHED
12148  046436  005012                  4$:     CLR     (R2)            ;DISABLE THE LCLK INTR
12149  046440  005703                          TST     R3              ;DID SOFTWARE FLAG GET SET ??
12150  046442  001404                          BEQ     8$              ;BR IF NOT - IT WORKED OK
12151  046444  104006                  3$:     ERROR   6               ;LCLK FAILED TO INTR ONTIME
12152  046446  000402                          BR      8$              ;GO EXIT
12153
12154  046450  005012                  6$:     CLR     (R2)            ;DISABLE  LCLK INTR
12155  046452  104006                  5$:     ERROR   6               ;LINE CLK TIMED OUT
12156
12157  046454  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12158  046456  012737  000102  000100          MOV     #102,@#100      ;RESTORE THE LINE CLK TRAPCATCHER
12159  046464  005037  000102                  CLR     @#102
12160
12161                                  ;;*****************************************************************
12162                                  ;*TEST 660      "BR6 VS BR4" PRIORITY ARBITRATION TEST
12163                                  ;THIS TEST VERIFIES THAT IF BOTH A "BR4" AND A "BR6" REQUEST ARE
12164                                  ;PENDING WHEN THE CPU PRIORITY IS LOWERED TO ALLOW INTRS. THAT "BR6"
12165                                  ;REQUEST IS GRANTED FIRST EVEN THOUGH THE "BR4" REQUEST MAY HAVE
12166                                  ;OCCURRED FIRST
12167                                  ;;*****************************************************************
12168  046470                          TST660:
12169  046470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12170  046472  012700  000660                  MOV     #660,R0         ;;LOAD R0 WITH TEST NUMBER
12171  046476  013701  046636                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12172  046502  010605                          MOV     SP,R5           ;SAVE THE SP
12173  046504  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12174  046510  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12175  046514  012703  177564                  MOV     #XCSR,R3        ;R3 POINTS TO DL11 XCSR
12176  046520  012737  046646  000100          MOV     #4$,@#100       ;SET UP THE LCLK VECTOR - GO TO 4$
12177  046526  012737  000300  000102          MOV     #300,@#102
12178  046534  012737  046700  000064          MOV     #8$,@#64        ;SET UP THE DL11 VECTOR - GO TO 8$
12179  046542  012737  000200  000066          MOV     #200,@#66
12180  046550  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12181  046552  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12182  046560  005037  063236                  CLR     @#MBUF0         ;INIT TIMER
12183  046564  005037  063242                  CLR     @#MBUF1         ;CLEAR DL11 INTR FLAG
12184  046570  005004                          CLR     R4              ;INIT TIMER
12185  046572  052713  000100                  BIS     #100,(R3)       ;ENABLE DL11 XMIT INTR
12186  046576  105713                  11$:    TSTB    (R3)            ;XMIT READY SET ??
12187  046600  100403                          BMI     12$             ;BR IF YES
12188  046602  005304                          DEC     R4              ;COUNT THE TIMER
12189  046604  001374                          BNE     11$             ;BR IF NO TIMEOUT
12190  046606  000443                          BR      5$              ;GO REPORT TIMEOUT FOR DL11
12191
12192  046610  005004                  12$:    CLR     R4              ;INIT THE TIMER AGAIN
12193  046612  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12194  046616  042712  000200                  BIC     #200,(R2)       ;CLEAR THE LINE CLOCK READY BIT
12195  046622  105712                  13$:    TSTB    (R2)            ;LCLK READY TO INTR
12196  046624  100403                          BMI     14$             ;BR IF YES
12197  046626  005304                          DEC     R4              ;COUNT THE TIMER
12198  046630  001374                          BNE     13$             ;BR IF NO TIMEOUT
12199  046632  000436                          BR      7$              ;GO REPORT LINE CLK TIMEOUT
```

```
12200  046634  000257                  14$:    CCC                     ;SCOPE SYNC
12201
12202  046636  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - KW SHOULD INTR FIRST
12203
12204  046642  005137  063236                  COM     @#MBUF0         ;SET SOFTWARE FLAG IF FETCHED
12205  046646  005013                  4$:     CLR     (R3)            ;DISABLE BOTH INTERRUPTS
12206  046650  005012                          CLR     (R2)
12207  046652  005737  063236                  TST     @#MBUF0         ;DID SOFTWARE FLAG GET SET ??
12208  046656  001402                          BEQ     6$              ;BR IF NOT
12209
12210  046660  104006                  3$:     ERROR   6               ;LINE CLK INTR OCCURRED TOO LATE
12211  046662  000425                          BR      9$              ;GO TO EXIT
12212
12213  046664  005737  063242          6$:     TST     @#MBUF1         ;DID DL11 SOFTWARE FLAG SET ??
12214  046670  001422                          BEQ     9$              ;BR IF NOT
12215
12216  046672  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12217  046674  104006                          ERROR   6               ;DL11 INTERRUPTED THE KW11
12218  046676  000417                          BR      9$              ;GO TO EXIT TEST
12219
12220  046700  005137  063242          8$:     COM     @#MBUF1         ;FLAG THE DL11 INTR
12221  046704  005013                          CLR     (R3)            ;DISABLE BOTH INTR ENABLES
12222  046706  005012                          CLR     (R2)
12223  046710  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12224  046712  104006                          ERROR   6               ;DL11 SHOULD NOT HAVE INTERRUPTED
12225  046714  000410                          BR      9$              ;GO EXIT TEST
12226
12227  046716  005012                  5$:     CLR     (R2)            ;DISABLE THE INTR ENABLES
12228  046720  005013                          CLR     (R3)
12229  046722  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12230  046724  104006                          ERROR   6               ;DL11 TIMEOUT
12231  046726  000403                          BR      9$              ;GO TO EXIT
12232
12233  046730  005012                  7$:     CLR     (R2)            ;DISABLE INTR ENABLES
12234  046732  005013                          CLR     (R3)
12235  046734  104006                          ERROR   6               ;KW11 TIMEOUT
12236
12237  046736  010506                  9$:     MOV     R5,SP           ;RESET THE SP
12238  046740  005037  177776                  CLR     @#PSW           ;RESET THE CPU PRIORITY
12239  046744  012737  000102  000100          MOV     #102,@#100      ;RESTORE LCLK VECTOR
12240  046752  005037  000102                  CLR     @#102
12241  046756  012737  000066  000064          MOV     #66,@#64        ;RESTORE THE DL11 XMIT VECTOR
12242  046764  005037  000066                  CLR     @#66
12243
12244                                  ; ***************************************************************************
12245                                  ; //////////////COMBINED INSTRUCTION EXERCISER SECTION //////////////
12246                                  ; ***************************************************************************
12247
12248                                  ;;*************************************************************
12249                                  ;*TEST 661       "BPT" TRAP LINKAGE TEST
12250                                  ;;*************************************************************
12251  046770                          T$T661:
12252  046770  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12253  046772  012700  000661                  MOV     #661,R0         ;;LOAD R0 WITH TEST NUMBER
12254  046776  013701  047022                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12255  047002  010605                          MOV     SP,R5           ;SAVE THE SP
```

```
12256  047004  010737  001010                        MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12257  047010  012737  047026  000014  1$:           MOV     #4$,@#14        ;GO TO 4$ ON "BPT" TRAP
12258  047016  010506                                MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12259  047020  000257                                CCC                     ;SCOPE SYNC
12260
12261  047022  000003                        2$:     BPT                     ;TEST THE "BPT" - GO TO 4$
12262
12263  047024  104005                        3$:     ERROR   5               ;BPT FAILED TO TRAP
12264
12265  047026  010506                        4$:     MOV     R5,SP           ;RESET THE SP
12266  047030  012737  000016  000014                MOV     #16,@#14        ;RESTORE THE VECTOR
12267
12268                                        ;;**************************************************************
12269                                        ;*TEST 662      RED ZONE OVERFLOW TEST - MOV R,-(SP)
12270                                        ;;**************************************************************
12271  047036                                TST662:
12272  047036  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12273  047040  012700  000662                        MOV     #662,R0         ;;LOAD R0 WITH TEST NUMBER
12274  047044  013701  047104                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12275  047050  010605                                MOV     SP,R5           ;SAVE SP
12276  047052  013704  000004                        MOV     @#4,R4          ;SAVE T.O. VECTOR
12277  047056  013703  000336                        MOV     @#336,R3        ;SAVE VECTOR AT 336
12278  047062  012737  047122  000004                MOV     #4$,@#4 ;GO TO 4$ ON OVFLW
12279  047070  012737  125252  000336                MOV     #125252,@#336   ;INIT. [336]
12280  047076  012706  000340                        MOV     #340,SP         ;SET SP TO CAUSE RED ZONE TRAP
12281  047102  000257                                CCC                     ;SCOPE SYNC
12282
12283  047104  010046                        2$:     MOV     R0,-(SP)        ;FORCE RED ZONE TRAP - GO TO 4$
12284
12285  047106  010437  000004                        MOV     R4,@#4          ;RESTORE T.O. VECTOR
12286  047112  010637  001074                        MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12287  047116  010506                                MOV     R5,SP           ;RESET SP FOR ERROR CALL
12288  047120  104005                        3$:     ERROR   5               ;MOV FAILED TO CAUSE TRAP
12289
12290  047122  010437  000004                4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
12291  047126  022706  000000                        CMP     #0,SP           ;[SP]=0?
12292  047132  001404                                BEQ     6$              ;BE IF YES
12293
12294  047134  010637  001074                        MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12295  047140  010506                                MOV     R5,SP           ;RESET SP FOR ERROR CALL
12296  047142  104005                        5$:     ERROR   5               ;SP NOT BEING JAMMED TO 4
12297
12298  047144  022737  125252  000336  6$:     CMP     #125252,@#336   ;DID PUSH OCCUR IN YELLOW ZONE?
12299  047152  001404                                BEQ     8$              ;BR IF NOT
12300
12301  047154  010637  001074                        MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12302  047160  010506                                MOV     R5,SP           ;RESET SP FOR ERROR CALL
12303  047162  104005                        7$:     ERROR   5               ;MOV PUSHED INTO YELLOW ZONE
12304
12305  047164  010337  000336                8$:     MOV     R3,@#336        ;RESTORE VECTOR 336
12306  047170  010506                                MOV     R5,SP           ;RESET SP
12307
12308                                        ;;**************************************************************
12309                                        ;*TEST 663      YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
12310                                        ;;**************************************************************
12311  047172                                TST663:
```

```
12312  047172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12313  047174  012700  000663                  MOV     #663,R0         ;;LOAD R0 WITH TEST NUMBER
12314  047200  013701  047232                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12315  047204  010605                          MOV     SP,R5           ;SAVE SP
12316  047206  012702  000376                  MOV     #376,R2         ;R2 POINTS TO STACK
12317  047212  013704  000004                  MOV     @#4,R4          ;SAVE T.O. VECTOR
12318  047216  012737  047250  000004          MOV     #4$,@#4 ;ON OVFLW - GO TO 4$
12319  047224  012706  000400                  MOV     #400,SP         ;SET SP TO CAUSE OVFLW
12320  047230  000257                          CCC                     ;SCOPE SYNC
12321
12322  047232  010046                  2$:     MOV     R0,-(SP)        ;FORCE STACK OVFLW - GO TO 4$
12323
12324  047234  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
12325  047240  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12326  047244  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
12327  047246  104005                  3$:     ERROR   5               ;STACK OVFLW FAILED TO TRAP
12328
12329  047250  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
12330  047254  020012                          CMP     R0,(R2)         ;DID [R0] GET PUSHED?
12331  047256  001404                          BEQ     6$              ;BR IF YES
12332
12333  047260  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12334  047264  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
12335  047266  104005                  5$:     ERROR   5               ;MOV FAILED TO PUSH IN YELLOW ZONE
12336
12337  047270  005706                  6$:     TST     SP              ;[SP]=0?
12338  047272  001004                          BNE     8$              ;BR IF NOT
12339
12340  047274  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12341  047300  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
12342  047302  104005                  7$:     ERROR   5               ;RED ZONE INSTEAD OF YELLOW ZONE
12343
12344  047304  010506                  8$:     MOV     R5,SP           ;RESET SP
12345
12346                                  ;;**************************************************************
12347                                  ;*TEST 664       YELLOW ZONE OVERFLOW TEST - (CMP  R0,-(SP))
12348                                  ;;**************************************************************
12349  047306                          TST664:
12350  047306  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12351  047310  012700  000664                  MOV     #664,R0         ;;LOAD R0 WITH TEST NUMBER
12352  047314  013701  047342                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12353  047320  010605                          MOV     SP,R5           ;SAVE THE SP
12354  047322  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
12355  047326  012737  047346  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
12356  047334  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
12357  047340  000257                          CCC                     ;SCOPE SYNC
12358
12359  047342  020046                  2$:     CMP     R0,-(SP)        ;TEST THE CMP - NO TRAP SHOULD OCCUR
12360
12361  047344  000406                          BR      6$              ;GO TO EXIT TEST
12362
12363  047346  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
12364  047352  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12365  047356  010506                          MOV     R5,SP           ;RESET THE SP
12366  047360  104005                  3$:     ERROR   5               ;CMP CAUSED OVERFLOW TRAP
12367
```

```
12368  047362  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
12369  047366  010506                          MOV     R5,SP           ;RESET THE SP
12370
12371                                  ;;*****************************************************************
12372                                  ;*TEST 665      YELLOW ZONE OVERFLOW TEST - (BIT  R0,-(SP))
12373                                  ;;*****************************************************************
12374  047370                          TST665:
12375  047370  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12376  047372  012700  000665                  MOV     #665,R0         ;;LOAD R0 WITH TEST NUMBER
12377  047376  013701  047424                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12378  047402  010605                          MOV     SP,R5           ;SAVE THE SP
12379  047404  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
12380  047410  012737  047430  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
12381  047416  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
12382  047422  000257                          CCC                     ;SCOPE SYNC
12383
12384  047424  030046                  2$:     BIT     R0,-(SP)        ;TEST THE BIT - NO TRAP SHOULD OCCUR
12385
12386  047426  000406                          BR      6$              ;GO TO EXIT TEST
12387
12388  047430  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
12389  047434  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12390  047440  010506                          MOV     R5,SP           ;RESET THE SP
12391  047442  104005                  3$:     ERROR   5               ;BIT CAUSED OVERFLOW TRAP
12392
12393  047444  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
12394  047450  010506                          MOV     R5,SP           ;RESET THE SP
12395
12396                                  ;;*****************************************************************
12397                                  ;*TEST 666      YELLOW ZONE OVERFLOW TEST - (TST  -(SP))
12398                                  ;;*****************************************************************
12399  047452                          TST666:
12400  047452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12401  047454  012700  000666                  MOV     #666,R0         ;;LOAD R0 WITH TEST NUMBER
12402  047460  013701  047506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12403  047464  010605                          MOV     SP,R5           ;SAVE THE SP
12404  047466  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
12405  047472  012737  047512  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
12406  047500  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
12407  047504  000257                          CCC                     ;SCOPE SYNC
12408
12409  047506  005746                  2$:     TST     -(SP)           ;TEST THE TST - NO TRAP SHOULD OCCUR
12410
12411  047510  000406                          BR      6$              ;GO TO EXIT TEST
12412
12413  047512  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
12414  047516  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12415  047522  010506                          MOV     R5,SP           ;RESET THE SP
12416  047524  104006                  3$:     ERROR   6               ;TST CAUSED OVERFLOW TRAP
12417
12418  047526  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
12419  047532  010506                          MOV     R5,SP           ;RESET THE SP
12420
12421                                  ;;*****************************************************************
12422                                  ;*TEST 667      ODD ADDRESS ERROR TEST - SUB  RA,(RB) - (RB) = ODD
12423                                  ;;*****************************************************************
```

```
12424  047534                                TST667:
12425  047534  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12426  047536  012700  000667                        MOV     #667,R0         ;;LOAD R0 WITH TEST NUMBER
12427  047542  013701  047576                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12428  047546  010605                              MOV     SP,R5           ;SAVE SP
12429  047550  010737  001010                        MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12430  047554  013704  000004              1$:       MOV     @#4,R4          ;SAVE T.O. VECTOR
12431  047560  012737  047606  000004              MOV     #4$,@#4 ;ON ODD ADDR ERROR - GO TO 4$
12432  047566  010506                              MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12433  047570  012702  000001                        MOV     #1,R2           ;R2 GETS ODD ADDRESS
12434  047574  000257                              CCC                     ;SCOPE SYNC
12435
12436  047576  160012                    2$:       SUB     R0,(R2)         ;FORCE ODD ADDR ERROR - GO TO 4$
12437
12438  047600  010437  000004                        MOV     R4,@#4          ;RESTORE T.O. VECTOR
12439  047604  104006                    3$:       ERROR   6               ;ODD ADDR FAILED TO TRAP
12440
12441  047606  010437  000004              4$:       MOV     R4,@#4          ;RESTORE T.O. VECTOR
12442  047612  010506                              MOV     R5,SP           ;RESET SP
12443  047614  005037  000000                        CLR     @#0             ;CLR LOC. 0 JUST IN CASE
12444
12445                                        ;;**************************************************************
12446                                        ;*TEST 670      TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
12447                                        ;;**************************************************************
12448  047620                                TST670:
12449  047620  000004                                SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12450  047622  012700  000670                        MOV     #670,R0         ;;LOAD R0 WITH TEST NUMBER
12451  047626  013701  047650                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12452  047632  012702  063243                        MOV     #MBUF1+1,R2     ;DEST ADDR=MBUF1+1 (ODD)
12453  047636  012737  047720  000004              MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
12454
12455  047644  010205                              MOV     R2,R5           ;[R5] = DEST. ADDR
12456  047646  000257                              CCC                     ;SCOPE SYNC
12457
12458  047650  105435                    2$:       NEGB    @(R5)+          ;TEST DM=3 TRAP
12459
12460  047652  104006                    3$:       ERROR   6               ;ODA TRAP NOT SPRUNG
12461
12462  047654  012705  063245                        MOV     #MBUF1+3,R5     ;[R5] = DEST. ADDR
12463  047660  013701  047666                        MOV     @#20$,R1        ;[R1] = TEST INSTR
12464  047664  000257                              CCC                     ;SCOPE SYNC
12465
12466  047666  105455                    20$:      NEGB    @-(R5)          ;TEST DM=5 TRAP
12467
12468  047670  104006                    5$:       ERROR   6               ;ODA TRAP NOT SPRUNG
12469
12470  047672  010205                              MOV     R2,R5           ;[R5] = DEST ADDR
12471  047674  013701  047702                        MOV     @#21$,R1        ;[R1] = TEST INSTR
12472  047700  000257                              CCC                     ;SCOPE SYNC
12473
12474  047702  105475  000000              21$:      NEGB    @0(R5)          ;TEST DM=7 TRAP
12475
12476  047706  104006                    7$:       ERROR   6               ;ODA TRAP NOT SPRUNG
12477
12478  047710  012737  061144  000004              MOV     #BERR,@#4       ;RESET T.O. VECTOR
12479  047716  000403                              BR      TST671          ;;GO TO SCOPE EXIT
```

```
12480
12481  047720  062716  000002          4$:     ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
12482  047724  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
12483
12484                                  ;;****************** ************************************************
12485                                  ;*TEST 671       TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
12486                                  ;;*************************************************************
12487  047726                          TST671:
12488  047726  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12489  047730  012700  000671                  MOV     #671,R0         ;;LOAD R0 WITH TEST NUMBER
12490  047734  013701  047756                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12491  047740  012702  063243                  MOV     #MBUF1+1,R2     ;[R2] = SOURCE ADDR. (ODD)
12492  047744  012737  050026  000004          MOV     #4$,@#4 ;GO TO 4$ ON TRAP
12493
12494  047752  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR.
12495  047754  000257                          CCC                     ;SCOPE SYNC
12496
12497  047756  113504                  2$:     MOVB    @(R5)+,R4       ;TEST SM=3
12498
12499  047760  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
12500
12501  047762  012705  063245                  MOV     #MBUF1+3,R5     ;[R5] = SOURCE ADDR
12502  047766  013701  047774                  MOV     @#20$,R1        ;[R1] = TEST INSTR
12503  047772  000257                          CCC                     ;SCOPE SYNC
12504
12505  047774  115504                  20$:    MOVB    @-(R5),R4       ;TEST SM=5
12506
12507  047776  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
12508  050000  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR
12509  050002  013701  050010                  MOV     @#21$,R1        ;[R1] = TEST INSTR
12510  050006  000257                          CCC                     ;SCOPE SYNC
12511
12512  050010  117504  000000          21$:    MOVB    @0(R5),R4       ;TEST SM=7
12513
12514  050014  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
12515
12516  050016  012737  061144  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
12517  050024  000403                          BR      TST672          ;;GO TO SCOPE EXIT
12518
12519  050026  062716  000002          4$:     ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
12520  050032  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
12521
12522                                  ;;*************************************************************
12523                                  ;*TEST 672       TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES
12524                                  ;;*************************************************************
12525  050034                          TST672:
12526  050034  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12527  050036  012700  000672                  MOV     #672,R0         ;;LOAD R0 WITH TEST NUMBER
12528  050042  013701  050064                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12529  050046  012702  050137                  MOV     #6$+3,R2        ;DEST ADDR = 6$+3 (ODD)
12530  050052  012737  050142  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
12531
12532  050060  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
12533  050062  000257                          CCC                     ;SCOPE SYNC
12534
12535  050064  000135                  2$:     JMP     @(R5)+          ;TEST JMP DM=3
```

```
12536
12537  050066  104006                  3$:   ERROR   6              ;ODA TRAP NOT SPRUNG IN ROM LOC 153
12538
12539  050070  012705  050137                MOV     #6$+3,R5       ;[R5] = DEST ADDR
12540  050074  013701  050102                MOV     @#20$,R1       ;[R1] = TEST INSTR
12541  050100  000257                        CCC                    ;SCOPE SYNC
12542
12543  050102  000155                 20$:   JMP     @-(R5)         ;TEST JMP DM=5
12544
12545  050104  104006                  5$:   ERROR   6              ;ODA TRAP NOT SPRUNG IN ROM LOC 155
12546
12547  050106  010205                        MOV     R2,R5          ;[R5] = DEST ADDR
12548  050110  013701  050116                MOV     @#21$,R1       ;[R1] = TEST INSTR
12549  050114  000257                        CCC                    ;SCOPE SYNC
12550
12551  050116  000175  000000         21$:   JMP     @0(R5)         ;TEST JMP DM=7
12552
12553  050122  104006                  7$:   ERROR   6              ;ODA TRAP NOT SPRUNG
12554
12555  050124  012737  061144  000004        MOV     #BERR,@#4      ;RESET BUS T.O. VECTOR
12556  050132  000420                        BR      TST673         ;;GO TO SCOPE EXIT
12557
12558  050134  000000                  6$:   HALT                   ;CATASTOPHIC ERROR - [PC] QUESTIONABLE.
12559  050136  000000                        HALT                   ;RESTART PROGRAM - DO NOT CONTINUE.
12560  050140  000000                        HALT
12561
12562  050142  032716  000001          4$:   BIT     #1,(SP)        ;TRAP DUE TO ODD PC?
12563  050146  001003                        BNE     8$             ;BR IF YES
12564  050150  062716  000002                ADD     #2,(SP)        ;MOV RETURN PC AROUND ERROR CALL
12565  050154  000002                        RTI                    ;RETURN TO NEXT SUB TEST
12566
12567  050156  011603                  8$:   MOV     (SP),R3        ;GET ODD PC OFF STACK INTO R3
12568  050160  062706  000004                ADD     #4,SP          ;FIX SP
12569
12570  050164  104007                  9$:   ERROR   7              ;PC TRAPPED WITH ODD ADDRESS
12571
12572  050166  012737  061144  000004        MOV     #BERR,@#4      ;RESET T.O. VECTOR
12573
12574                                 ;;*****************************************************************
12575                                 ;*TEST 673      TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
12576                                 ;;*****************************************************************
12577  050174                         TST673:
12578  050174  000004                        SCOPE                  ;CALL THE SCOPE LOOP UTILITY
12579  050176  012700  000673                MOV     #673,R0        ;;LOAD R0 WITH TEST NUMBER
12580  050202  013701  050226                MOV     @#2$,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
12581  050206  012737  050340  000004        MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
12582  050214  010605                        MOV     SP,R5          ;SAVE SP
12583  050216  012702  000376                MOV     #376,R2        ;USE R2 TO SET UP SP TO CAUSE TRAP
12584
12585  050222  010206                        MOV     R2,SP          ;SET UP SP TO CAUSE OVERFLOW
12586  050224  000257                        CCC                    ;SCOPE SYNC
12587
12588  050226  005016                  2$:   CLR     (SP)           ;TEST DM1 - SHOULD SPRING TRAP
12589
12590  050230  010637  001074                MOV     SP,@#$REGS     ;SAVE BAD SP FOR PRINTING
12591  050234  010506                        MOV     R5,SP          ;RESET SP
```

```
12592  050236  104006                  3$:     ERROR   6               ;DM1 FAILED TO CAUSE OVERFLOW TRAP
12593
12594  050240  013701  050250                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
12595  050244  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12596  050246  000257                          CCC                     ;SCOPE SYNC
12597
12598  050250  005026                  20$:    CLR     (SP)+           ;TEST DM2 - SHOULD SPRING TRAP
12599
12600  050252  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12601  050256  010506                          MOV     R5,SP           ;RESET SP
12602  050260  104006                  5$:     ERROR   6               ;DM2 FAILED TO CAUSE OVERFLOW TRAP
12603
12604  050262  013701  050272                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
12605  050266  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12606  050270  000257                          CCC                     ;SCOPE SYNC
12607
12608  050272  005046                  21$:    CLR     -(SP)           ;TEST DM4 - SHOULD SPRING TRAP
12609
12610  050274  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12611  050300  010506                          MOV     R5,SP           ;RESET SP
12612  050302  104006                  7$:     ERROR   6               ;DM4 FAILED TO CAUSE OVERFLOW TRAP
12613
12614  050304  013701  050314                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
12615  050310  010206                          MOV     R2,SP           ;SET SP TO CAUSE ERROR
12616  050312  000257                          CCC                     ;SCOPE SYNC
12617
12618  050314  005066  000000          22$:    CLR     0(SP)           ;TEST DM6 - SHOULD SPRING TRAP
12619
12620  050320  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12621  050324  010506                          MOV     R5,SP           ;RESET SP
12622  050326  104006                  9$:     ERROR   6               ;DM6 FAILED TO CAUSE OVERFLOW TRAP
12623
12624  050330  012737  061144  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
12625  050336  000407                          BR      TST674          ;;GO TO SCOPE EXIT
12626
12627  050340  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC OFF STACK
12628  050342  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
12629  050346  010506                          MOV     R5,SP           ;RESET SP
12630  050350  005046                          CLR     -(SP)           ;PUSH NEW PS ON STACK
12631  050352  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC ON STACK
12632  050354  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
12633
12634                                  ;;*****************************************************************
12635                                  ;*TEST 674       TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
12636                                  ;;*****************************************************************
12637  050356                          TST674:
12638  050356  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12639  050360  012700  000674                  MOV     #674,R0         ;;LOAD R0 WITH TEST NUMBER
12640  050364  013701  050410                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12641  050370  012737  050522  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLW TRAP
12642  050376  010605                          MOV     SP,R5           ;SAVE SP
12643  050400  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
12644
12645  050404  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12646  050406  000257                          CCC                     ;SCOPE SYNC
12647
```

```
12648  050410  010016                  2$:     MOV     R0,(SP)         ;TEST MOV DM1 - SHOULD SPRING TRAP
12649
12650  050412  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12651  050416  010506                          MOV     R5,SP           ;RESET SP
12652  050420  104006                  3$:     ERROR   6               ;MOV DM1 FAILED TO SPRING TRAP
12653
12654  050422  013701  050432                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
12655  050426  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12656  050430  000257                          CCC                     ;SCOPE SYNC
12657
12658  050432  010026                  20$:    MOV     R0,(SP)+        ;TEST MOV DM2 - SHOULD SPRING TRAP
12659
12660  050434  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12661  050440  010506                          MOV     R5,SP           ;RESET SP
12662  050442  104006                  5$:     ERROR   6               ;MOV DM2 FAILED TO SPRING TRAP
12663
12664  050444  013701  050454                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
12665  050450  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12666  050452  000257                          CCC                     ;SCOPE SYNC
12667
12668  050454  010046                  21$:    MOV     R0,-(SP)        ;TEST MOV DM4 - SHOULD SPRING TRAP
12669
12670  050456  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12671  050462  010506                          MOV     R5,SP           ;RESET SP
12672  050464  104006                  7$:     ERROR   6               ;MOV DM4 FAILED TO SPRING TRAP
12673
12674  050466  013701  050476                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
12675  050472  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
12676  050474  000257                          CCC                     ;SCOPE SYNC
12677
12678  050476  010066  000000          22$:    MOV     R0,0(SP)        ;TEST MOV DM6 - SHOULD SPRING TRAP
12679
12680  050502  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR PRINTING
12681  050506  010506                          MOV     R5,SP           ;RESET SP
12682  050510  104006                  9$:     ERROR   6               ;MOV DM6 FAILED TO CAUSE OVFLW TRAP
12683
12684  050512  012737  061144  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
12685  050520  000407                          BR      TST675          ;;GO TO SCOPE EXIT
12686
12687  050522  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC
12688  050524  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
12689  050530  010506                          MOV     R5,SP           ;RESET SP
12690  050532  005046                          CLR     -(SP)           ;PUSH NEW PSW
12691  050534  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC
12692  050536  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
12693
12694                           ;;**************************************************************
12695                           ;*TEST 675      TEST THAT JSR CAN CAUSE OVERFLOW TRAP
12696                           ;;**************************************************************
12697  050540                  TST675:
12698  050540  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12699  050542  012700  000675                  MOV     #675,R0         ;;LOAD R0 WITH TEST NUMBER
12700  050546  013701  050570                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12701  050552  012737  050612  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVERFLOW ERROR
12702  050560  010605                          MOV     SP,R5           ;SAVE SP
12703  050562  012706  000400                  MOV     #400,SP         ;SET THE SP TO CAUSE TRAP
```

```
12704  050566  000257                          CCC                     ;SCOPE SYNC
12705
12706  050570  004737  050616          2$:     JSR     PC,@#6$ ;TEST JSR - SHOULD SPRING TRAP
12707
12708  050574  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
12709  050600  010506                          MOV     R5,SP           ;RESET SP
12710  050602  104005                  3$:     ERROR   5               ;JSR PUSH DID NOT SPRING OVFL TRAP
12711
12712  050604  000410                          BR      8$              ;GO TO SCOPE EXIT
12713
12714  050606  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
12715  050612  010506                  4$:     MOV     R5,SP           ;RESET SP
12716  050614  000404                          BR      8$              ;GO EXIT TEST - ALL OK
12717
12718  050616  010637  001074          6$:     MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
12719  050622  010506                          MOV     R5,SP           ;RESET SP
12720  050624  104005                  5$:     ERROR   5               ;JSR PUSH FAILED TO SPRING OVFLW TRAP
12721
12722  050626  012737  061144  000004  8$:     MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
12723
12724                                  ;;*****************************************************************
12725                                  ;*TEST 676      TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
12726                                  ;;*****************************************************************
12727  050634                          TST676:
12728  050634  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12729  050636  012700  000676                  MOV     #676,R0         ;;LOAD R0 WITH TEST NUMBER
12730  050642  013701  050676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12731  050646  013704  000014                  MOV     @#14,R4         ;SAVE BREAK POINT TRAP VECTOR
12732  050652  010605                          MOV     SP,R5           ;SAVE SP
12733  050654  012737  050712  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
12734  050662  012737  050716  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
12735  050670  012706  000400                  MOV     #400,SP         ;SET UP SP TO CAUSE OVFLW ON 1ST PUSH
12736  050674  000257                          CCC                     ;SCOPE SYNC
12737
12738  050676  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
12739
12740  050700  010637  001074                  MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
12741  050704  010506                          MOV     R5,SP           ;RESET SP
12742  050706  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
12743
12744  050710  000406                          BR      8$              ;GO TO SCOPE EXIT
12745
12746  050712  010506                  4$:     MOV     R5,SP           ;RESET SP
12747  050714  000404                          BR      8$              ;GO EXIT - ALL OK
12748
12749  050716  010637  001074          6$:     MOV     SP,@#$REGS      ;SAVE BAD SP FOR ERROR PRINTOUT
12750  050722  010506                          MOV     R5,SP           ;RESET SP
12751  050724  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
12752
12753  050726  012737  061144  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
12754  050734  010437  000014                  MOV     R4,@#14
12755
12756                                  ;;*****************************************************************
12757                                  ;*TEST 677      TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
12758                                  ;;*****************************************************************
12759  050740                          TST677:
```

```
12760  050740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12761  050742  012700  000677                  MOV     #677,R0         ;;LOAD R0 WITH TEST NUMBER
12762  050746  013701  051002                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12763  050752  013704  000014                  MOV     @#14,R4         ;SAVE BPT VECTOR
12764  050756  010605                          MOV     SP,R5           ;SAVE SP
12765  050760  012737  051016  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLOW
12766  050766  012737  051022  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
12767  050774  012706  000402                  MOV     #402,SP         ;SET SP TO CAUSE TRAP ON 2ND PUSH
12768  051000  000257                          CCC                     ;SCOPE SYNC
12769
12770  051002  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
12771
12772  051004  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12773  051010  010506                          MOV     R5,SP           ;RESET SP
12774  051012  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
12775
12776  051014  000406                          BR      8$              ;GO TO SCOPE EXIT
12777
12778  051016  010506                  4$:     MOV     R5,SP           ;RESET SP
12779  051020  000404                          BR      8$              ;GO EXIT - ALL OK
12780
12781  051022  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12782  051026  010506                          MOV     R5,SP           ;RESET SP
12783  051030  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
12784
12785  051032  012737  061144  000004  8$:     MOV     #8ERR,@#4       ;RESET VECTORS
12786  051040  010437  000014                  MOV     R4,@#14
12787
12788                                  ;;*****************************************************************
12789                                  ;*TEST 700       ILLEGAL INSTRUCTION TEST - JSR RN,%R
12790                                  ;;*****************************************************************
12791  051044                          ST700:
12792  051044  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12793  051046  012700  000700                  MOV     #700,R0         ;;LOAD R0 WITH TEST NUMBER
12794  051052  013701  051106                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12795  051056  010605                          MOV     SP,R5           ;SAVE SP
12796  051060  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12797  051064  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
12798  051070  012737  051116  000004          MOV     #4$,@#4 ;ILLEGAL INSTR. TRAP GOES TO 4$
12799  051076  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12800  051100  012702  051114                  MOV     #3$,R2          ;IN CASE JSR JUMPS TO [R2]
12801  051104  000257                          CCC                     ;SCOPE SYNC
12802
12803  051106  004302                  2$:     JSR     R3,R2           ;JSR MODE 0 FORCES TRAP - GO TO 4$
12804
12805  051110  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
12806  051114  104005                  3$:     ERROR   5               ;JSR FAILED TO SPRING TRAP
12807
12808  051116  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR
12809  051122  010506                          MOV     R5,SP           ;RESET SP
12810
12811                                  ;;*****************************************************************
12812                                  ;*TEST 701       ILLEGAL INSTRUCTION TEST - JMP %R
12813                                  ;;*****************************************************************
12814  051124                          ST701:
12815  051124  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
12816  051126  012700  000701                      MOV     #701,R0         ;;LOAD R0 WITH TEST NUMBER
12817  051132  013701  051166                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12818  051136  010605                              MOV     SP,R5           ;SAVE SP
12819  051140  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12820  051144  013704  000004              1$:     MOV     @#4,R4          ;SAVE VECTOR POINTER AT LOC. 4
12821  051150  012737  051176  000004              MOV     #4$,@#4 ;ON TRAP - GO TO 4$
12822  051156  010506                              MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12823  051160  012702  051174                      MOV     #3$,R2          ;IN CASE IT JUMPS TO ADDR IN RN
12824  051164  000257                              CCC                     ;SCOPE SYNC
12825
12826  051166  000102              2$:     JMP     R2              ;JMP MODE 0 FORCES TRAP - GO TO 4$
12827
12828  051170  010437  000004                      MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
12829  051174  104005              3$:     ERROR   5               ;ILLEGAL INSTR TRAP FAILED
12830
12831  051176  010437  000004      4$:     MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
12832  051202  010506                      MOV     R5,SP           ;RESET SP
12833
12834                               ;;**********************************************************************
12835                               ;*TEST 702      BUS TIMEOUT TRAP TEST - TST (R)
12836                               ;;**********************************************************************
12837  051204                      TST702:
12838  051204  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12839  051206  012700  000702                      MOV     #702,R0         ;;LOAD R0 WITH TEST NUMBER
12840  051212  013701  051246                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12841  051216  010605                              MOV     SP,R5           ;SAVE SP
12842  051220  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12843  051224  013704  000004              1$:     MOV     @#4,R4          ;SAVE ORIGINAL T.O. VECTOR POINTER
12844  051230  012737  051256  000004              MOV     #4$,@#4 ;ON T.O. TRAP - GO TO 4$
12845  051236  012702  160000                      MOV     #160000,R2      ;ADDRESS CAUSES T.O.
12846  051242  010506                              MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12847  051244  000257                              CCC                     ;SCOPE SYNC
12848
12849  051246  005712              2$:     TST     (R2)            ;FORCE T.O. TRAP - GO TO 4$
12850
12851  051250  010437  000004                      MOV     R4,@#4          ;RESTORE T.O. VECTOR
12852  051254  104005              3$:     ERROR   5               ;TIMEOUT TRAP FAILED
12853  051256  010437  000004      4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
12854  051262  010506                      MOV     R5,SP           ;RESET SP
12855
12856                               ;;**********************************************************************
12857                               ;*TEST 703      "T" BIT TRAP TEST
12858                               ;;**********************************************************************
12859  051264                      TST703:
12860  051264  000004                              SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12861  051266  012700  000703                      MOV     #703,R0         ;;LOAD R0 WITH TEST NUMBER
12862  051272  013701  051330                      MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12863  051276  010605                              MOV     SP,R5           ;SAVE SP
12864  051300  010737  001010                      MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12865  051304  010506              1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12866  051306  012737  051336  000014              MOV     #4$,@#14        ;GO TO 4$ WHEN "T" TRAP SPRUNG
12867  051314  012746  000020                      MOV     #20,-(SP)       ;SET "T" BIT ON STACK
12868  051320  012746  051330                      MOV     #2$,-(SP)       ;SET UP NEW PC ON STACK
12869  051324  000257                              CCC                     ;SCOPE SYNC
12870  051326  000006                              RTT                     ;TURN ON "T" BIT - GO TO 2$
12871
```

```
12872  051330  005700                  2$:     TST     R0              ;SPRING "T" BIT TRAP - GO TO 4$
12873
12874  051332  104005                  3$:     ERROR   5               ;NO "T" BIT TRAP OCCURRED
12875
12876  051334  000405                          BR      6$              ;GO EXIT
12877
12878  051336  032766  000020  000002  4$:     BIT     #20,2(SP)       ;"T" BIT SET IN OLD PSW?
12879  051344  001001                          BNE     6$              ;BR IF YES
12880
12881  051346  104001                  5$:     ERROR   1               ;#T# BIT NOT SAVED ON STACK
12882
12883  051350  012737  000016  000014  6$:     MOV     #16,@#14        ;RESTORE "T" BIT TRAP CATCHER
12884  051356  005037  000016                  CLR     @#16
12885  051362  010506                          MOV     R5,SP           ;RESET SP
12886
12887                                  ;;**********************************************************************
12888                                  ;*TEST 704      TEST PUSH INTO PSW WITH (SP) = 000000
12889                                  ;THESE NEXT TWO TESTS VERIFY THAT A "RED ZONE" TRAP OCCURS IF A
12890                                  ;PUSH IS ATTEMPTED WITH THE (SP) INITIALLY EQUAL TO 000000,177572,
12891                                  ;;**********************************************************************
12892  051364                          TST704:
12893  051364  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12894  051366  012700  000704                  MOV     #704,R0         ;;LOAD R0 WITH TEST NUMBER
12895  051372  013701  051416                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12896  051376  010605                          MOV     SP,R5           ;SAVE THE SP
12897  051400  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
12898  051404  012737  051440  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
12899  051412  005006                          CLR     SP              ;MAKE SP = 000000
12900  051414  000257                          CCC                     ;SCOPE SYNC
12901
12902  051416  012746  007777          2$:     MOV     #7777,-(SP)     ;ATTEMPT PUSH INTO PSW - SHOULD CAUSE
12903                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
12904
12905  051422  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12906  051426  005004                          CLR     R4              ;(R4) = S / B SP
12907  051430  010603                          MOV     SP,R3           ;(R3) = WAS SP
12908  051432  010506                          MOV     R5,SP           ;RESET THE SP
12909  051434  104003                  3$:     ERROR   3               ;TRAP NOT SPRUNG
12910  051436  000414                          BR      TST705          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
12911
12912  051440  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
12913  051444  001406                          BEQ     6$              ;BR IF YES
12914
12915  051446  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12916  051452  005004                          CLR     R4              ;(R4)= S / B SP
12917  051454  010603                          MOV     SP,R3           ;(R3) = WAS SP
12918  051456  010506                          MOV     R5,SP           ;RESET THE SP
12919  051460  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
12920
12921  051462  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
12922  051464  010437  000004                  MOV     R4,@#4          ;RESTORE BERR VECTOR
12923
12924                                  ;;**********************************************************************
12925                                  ;*TEST 705      TEST PUSH INTO SR WITH (SP) = 177572
12926                                  ;;**********************************************************************
12927  051470                          TST705:
```

```
12928  051470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12929  051472  012700  000705                  MOV     #705,R0         ;;LOAD R0 WITH TEST NUMBER
12930  051476  013701  051524                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12931  051502  010605                          MOV     SP,R5           ;SAVE THE SP
12932  051504  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
12933  051510  012737  051546  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
12934  051516  012706  177572                  MOV     #177572,SP      ;MAKE SP=177572
12935  051522  000257                          CCC                     ;SCOPE SYNC
12936
12937  051524  012746  177777          2$:     MOV     #-1,-(SP)       ;ATTEMPT PUSH INTO SR - SHOULD CAUSE
12938                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
12939
12940  051530  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12941  051534  005004                          CLR     R4              ;[R4] = S / B SP
12942  051536  010603                          MOV     SP,R3           ;[R3] = WAS SP
12943  051540  010506                          MOV     R5,SP           ;RESET THE SP
12944  051542  104003          3$:     ERROR   3                       ;TRAP NOT SPRUNG
12945  051544  000414                          BR      TST706          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
12946
12947  051546  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
12948  051552  001406                          BEQ     6$              ;BR IF YES
12949
12950  051554  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12951  051560  005004                          CLR     R4              ;[R4]= S / B SP
12952  051562  010603                          MOV     SP,R3           ;[R3] = WAS SP
12953  051564  010506                          MOV     R5,SP           ;RESET THE SP
12954  051566  104003          5$:     ERROR   3                       ;TRAP SPRUNG BUT NOT RED ZONE
12955
12956  051570  010506          6$:     MOV     R5,SP           ;FIX UP THE SP
12957  051572  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12958
12959                          ;;*****************************************************************
12960                          ;*TEST 706      TEST PUSH INTO SLR WITH [SP] = 177776
12961                          ;;*****************************************************************
12962  051576                  TST706:
12963  051576  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12964  051600  012700  000706                  MOV     #706,R0         ;;LOAD R0 WITH TEST NUMBER
12965  051604  013701  051632                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12966  051610  010605                          MOV     SP,R5           ;SAVE THE SP
12967  051612  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
12968  051616  012737  051654  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
12969  051624  012706  177776                  MOV     #177776,SP      ;MAKE SP=177776
12970  051630  000257                          CCC                     ;SCOPE SYNC
12971
12972  051632  012746  000200          2$:     MOV     #200,-(SP)      ;ATTEMPT PUSH INTO SLR - SHOULD CAUSE
12973                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
12974
12975  051636  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12976  051642  005004                          CLR     R4              ;[R4] = S / B SP
12977  051644  010603                          MOV     SP,R3           ;[R3] = WAS SP
12978  051646  010506                          MOV     R5,SP           ;RESET THE SP
12979  051650  104003          3$:     ERROR   3                       ;TRAP NOT SPRUNG
12980  051652  000414                          BR      TST707          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
12981
12982  051654  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
12983  051660  001406                          BEQ     6$              ;BR IF YES
```

```
12984
12985  051662  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12986  051666  005004                          CLR     R4              ;(R4)= S / B SP
12987  051670  010603                          MOV     SP,R3           ;(R3) = WAS SP
12988  051672  010506                          MOV     R5,SP           ;RESET THE SP
12989  051674  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
12990
12991  051676  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
12992  051700  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
12993
12994                                  ;;*****************************************************************
12995                                  ;*TEST 707      RSVD INSTRUCTION TEST - 000007 THRU 000077
12996                                  ;;*****************************************************************
12997  051704                          TST707:
12998  051704  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12999  051706  012700  000707                  MOV     #707,R0         ;;LOAD R0 WITH TEST NUMBER
13000  051712  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13001  051714  012737  051752  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13002  051722  005037  000012                  CLR     @#12
13003  051726  012701  000007                  MOV     #7,R1           ;SET UP FIRST ONE IN GROUP
13004  051732  010737  001010                  MOV     PC,@#$LPERR     ;ONLY LOOP ON BAD OP CODE
13005  051736  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13006  051740  010137  051746                  MOV     R1,@#2$ ;LOAD NEW INSTR
13007  051744  000257                          CCC                     ;SCOPE SYNC
13008
13009  051746  000007                  2$:     000007                  ;TEST THE RSVD INSTR - THIS LOCATION
13010                                                                   ;GETS CHANGED EACH PASS THROUGH
13011
13012  051750  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13013
13014  051752  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13015  051754  022701  000100                  CMP     #100,R1         ;AT END OF THIS GROUP ??
13016  051760  001366                          BNE     1$              ;BR IF NOT
13017
13018  051762  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13019  051764  012737  051712  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13020                                  ;;*****************************************************************
13021                                  ;*TEST 710      RSVD INSTRUCTION TEST - 000210 THRU 000237
13022                                  ;;*****************************************************************
13023  051772                          TST710:
13024  051772  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13025  051774  012700  000710                  MOV     #710,R0         ;;LOAD R0 WITH TEST NUMBER
13026  052000  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13027  052002  012737  052040  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13028  052010  005037  000012                  CLR     @#12
13029  052014  012701  000210                  MOV     #210,R1         ;SET UP FIRST ONE IN GROUP
13030  052020  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13031  052024  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13032  052026  010137  052034                  MOV     R1,@#2$ ;LOAD NEW INSTR
13033  052032  000257                          CCC                     ;SCOPE SYNC
13034
13035  052034  000210                  2$:     000210                  ;TEST THE RSVD INSTR - THIS LOCATION
13036                                                                   ;GETS CHANGED EACH PASS THROUGH
13037
13038  052036  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13039
```

```
13040  052040  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13041  052042  022701  000240                  CMP     #240,R1         ;AT END OF THIS GROUP ??
13042  052046  001366                          BNE     1$              ;BR IF NOT
13043
13044  052050  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13045  052052  012737  052000  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13046
13047                                  ;;*****************************************************************
13048                                  ;*TEST 711      RSVD INSTRUCTION TEST - 007000 THRU 007777
13049                                  ;*****************************************************************
13050  052060                          TST711:
13051  052060  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13052  052062  012700  000711                  MOV     #711,R0         ;;LOAD R0 WITH TEST NUMBER
13053  052066  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13054  052070  012737  052126  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13055  052076  005037  000012                  CLR     @#12
13056  052102  012701  007000                  MOV     #7000,R1        ;SET UP FIRST ONE IN GROUP
13057  052106  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13058  052112  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13059  052114  010137  052122                  MOV     R1,@#2$ ;LOAD NEW INSTR
13060  052120  000257                          CCC                     ;SCOPE SYNC
13061
13062  052122  007000                  2$:     007000                  ;TEST THE RSVD INSTR - THIS LOCATION
13063                                                                  ;GETS CHANGED EACH PASS THROUGH
13064
13065  052124  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13066
13067  052126  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13068  052130  022701  010000                  CMP     #10000,R1       ;AT END OF THIS GROUP ??
13069  052134  001366                          BNE     1$              ;BR IF NOT
13070
13071  052136  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13072  052140  012737  052066  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13073
13074                                  ;;*****************************************************************
13075                                  ;*TEST 712      RSVD INSTRUCTION TEST - 075000 THRU 076777
13076                                  ;*****************************************************************
13077  052146                          TST712:
13078  052146  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13079  052150  012700  000712                  MOV     #712,R0         ;;LOAD R0 WITH TEST NUMBER
13080  052154  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13081  052156  012737  052216  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13082  052164  005037  000012                  CLR     @#12
13083  052170  012701  075000                  MOV     #75000,R1       ;SET UP FIRST ONE IN GROUP
13084  052174  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13085  052200  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13086  052202  010137  052210                  MOV     R1,@#2$ ;LOAD NEW INSTR
13087  052206  000257                          CCC                     ;SCOPE SYNC
13088
13089  052210  075000                  2$:     75000                   ;TEST THE RSVD INSTR - THIS LOCATION
13090                                                                  ;GETS CHANGED EACH PASS THROUGH
13091
13092  052212  000240                          NOP                     ;IN CASE NON TRAPPING INSTR IS TWO WORDS
13093  052214  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13094
13095  052216  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
```

```
13096  052220  022701  076600                 CMP     #MFD,R1          ;MFD INSTRUCTION?
13097  052224  001774                         BEQ     4$               ;BR IF YES--SKIP IT.
13098  052226  022701  077000                 CMP     #077000,R1       ;AT END OF THIS GROUP ??
13099  052232  001362                         BNE     1$               ;BR IF NOT
13100
13101  052234  010506                         MOV     R5,SP            ;MAKE SURE TO RESET THE SP
13102  052236  012737  052154  001010         MOV     #5$,@#$LPERR     ;LOOP FROM BEGINNING ON ERROR
13103
13104                                 ;;*****************************************************************
13105                                 ;*TEST 713      RSVD INSTRUCTION TEST - 106400 THRU 107777
13106                                 ;;*********************************************** *****************
13107  052244                         TST713:
13108  052244  000004                         SCOPE                    ;CALL THE SCOPE LOOP UTILITY
13109  052246  012700  000713                 MOV     #713,R0          ;;LOAD R0 WITH TEST NUMBER
13110  052252  010605                 5$:     MOV     SP,R5            ;SAVE THE SP
13111  052254  012737  052312  000010         MOV     #4$,@#10         ;SET UP RSVD INSTR. TRAP VECTOR
13112  052262  005037  000012                 CLR     @#12
13113  052266  012701  106400                 MOV     #106400,R1       ;SET UP FIRST ONE IN GROUP
13114  052272  010737  001010                 MOV     PC,@#$LPERR      ;SET ERROR LOOP ADDRESS
13115  052276  010506                 1$:     MOV     R5,SP            ;RESET SP FOR ERROR LOOP AND NEW INSTR
13116  052300  010137  052306                 MOV     R1,@#2$ ;LOAD NEW INSTR
13117  052304  000257                         CCC                      ;SCOPE SYNC
13118
13119  052306  106400                 2$:     106400                   ;TEST THE RSVD INSTR - THIS LOCATION
13120                                                                   ;GETS CHANGED EACH PASS THROUGH
13121
13122  052310  104005                 3$:     ERROR   5                ;RSVD INSTR. IN R1 FAILED TO TRAP
13123
13124  052312  005201                 4$:     INC     R1               ;GENERATE NEW RSVD INSTR
13125  052314  022701  106500                 CMP     #106500,R1       ;MFPD INSTRUCTION ??
13126  052320  001002                         BNE     10$              ;BR IF NOT
13127  052322  012701  106700                 MOV     #106700,R1       ;SKIP MFPD AND MTPD INSTRUCTIONS
13128  052326  022701  110000         10$:    CMP     #110000,R1       ;AT END OF THIS GROUP ??
13129  052332  001361                         BNE     1$               ;BR IF NOT
13130
13131  052334  010506                         MOV     R5,SP            ;MAKE SURE TO RESET THE SP
13132  052336  012737  052252  001010         MOV     #5$,@#$LPERR     ;LOOP FROM BEGINNING ON ERROR
13133  052344  012737  061046  000010         MOV     #RSERR,@#10      ;RESTORE RSVD INSTR VECTOR
13134  052352  012737  000340  000012         MOV     #340,@#12
13135  052360  000004                         SCOPE                    ;CALL THE SCOPE LOOP UTILITY
13136
13137                                 ;THIS NEXT GROUP OF SEQUENTIAL TESTS VERIFIES THAT A "T" BIT
13138                                 ;TRAP CAN BE SERVICED IN EACH MICROWORD THAT DOES A "BUT SERVICE"
13139                                 ;EACH ROUTINE ENTERS THE TRAP MICROUTINE WHEN THE TRAP IS SPRUNG
13140
13141  052362  012737  061014  000014 TSET:    MOV     #TBSER,@#14      ;SET UP THE "T" BIT TRAP VECTOR
13142  052370  012737  000340  000016         MOV     #340,@#16        ;PRIORITY 7
13143
13144                                 ;;*****************************************************************
13145                                 ;*TEST 714      BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
13146                                 ;"INSTAB" (INSTRUCTION TABLE) CONTAINS ALL ONE WORD INSTRUCTIONS
13147                                 ;THAT TEST A "BUT SERVICE" IN A UNIQUE ROM LOCATION.  THE TABLE MUST
13148                                 ;BE TERMINATED WITH A 0 ENTRY.
13149                                 ;;*****************************************************************
13150  052376                         TST714:
13151  052376  012700  000714                 MOV     #714,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
13152  052402  010605                    6$:   MOV   SP,R5          ;SAVE THE SP
13153  052404  012704  063562                  MOV   #INSTAB,R4     ;PUT POINTER TO TABLE IN R4
13154  052410  012401                    4$:   MOV   (R4)+,R1       ;LOAD R1 WITH TEST INSTRUCTION WORD
13155  052412  001422                          BEQ   5$             ;EXIT TEST IF END OF TABLE
13156  052414  010737  001010                  MOV   PC,@#$LPERR    ;LOOP ON FAILING INSTRUCTION ONLY
13157  052420  010137  052452              1$:   MOV   R1,@#2$      ;STORE TEST INSTRUCTION TO BE EXECUTED
13158  052424  012702  063236                  MOV   #MBUF0,R2      ;IN CASE DM1 DEST--(R2)
13159  052430  012703  063242                  MOV   #MBUF1,R3      ;IN CASE SM1--(R3)
13160  052434  010506                          MOV   R5,SP          ;RESTORE SP FOR ERROR LOOPING
13161  052436  012746  000020                  MOV   #20,-(SP)      ;SET "T" BIT IN THE NEW PSW
13162  052442  012746  052452                  MOV   #2$,-(SP)      ;MAKE NEW PC = 2$
13163  052446  000257                          CCC                  ;SCOPE SYNC
13164  052450  000006                          RTT                  ;SET "T" BIT - GO TO 2$
13165
13166  052452  000240                    2$:   NOP                  ;INSTRUCTION FROM TABLE IS STORED HERE AND
13167                                                               ;SHOULD SPRING TRAP
13168
13169  052454  104005                    3$:   ERROR 5              ;BUT SERVICE FAILED
13170
13171  052456  000754                          BR    4$             ;GET NEXT INSTRUCTION FOR BUT SERVICE TEST
13172  052460  012737  052402  001010    5$:   MOV   #6$,@#$LPERR  ;LOOP FROM BEGINNING ON ERROR
13173
13174                                    ;;*****************************************************************
13175                                    ;*TEST 715      BUT SERVICE TEST - (RTI)
13176                                    ;;*****************************************************************
13177  052466                            TST715:
13178  052466  000004                          SCOPE                ;CALL THE SCOPE LOOP UTILITY
13179  052470  012700  000715                  MOV   #715,R0        ;;LOAD R0 WITH TEST NUMBER
13180  052474  013701  052512                  MOV   @#2$,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13181  052500  012746  000020                  MOV   #20,-(SP)      ;SET "T" BIT IN THE NEW PSW
13182  052504  012746  052514                  MOV   #3$,-(SP)      ;MAKE NEW PC = 3$
13183  052510  000257                          CCC                  ;SCOPE SYNC
13184
13185  052512  000002                    2$:   RTI                  ;INSTRUCTION SHOULD SPRING TRAP
13186
13187  052514  104005                    3$:   ERROR 5              ;BUT SERVICE IN XXX FAILED
13188
13189                                    ;;*****************************************************************
13190                                    ;*TEST 716      BUT SERVICE TEST - (JSR  %R,@A)
13191                                    ;;*****************************************************************
13192  052516                            TST716:
13193  052516  000004                          SCOPE                ;CALL THE SCOPE LOOP UTILITY
13194  052520  012700  000716                  MOV   #716,R0        ;;LOAD R0 WITH TEST NUMBER
13195  052524  013701  052574                  MOV   @#2$,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13196                                    .SBTTL USER CONTROLLED BREAKPOINT -- BIT14
13197  052530  032737  040000  063160          BIT   #BIT14,@#BPTLOC ;BREAKPOINT HALT SET ??
13198  052536  001401                          BEQ   .+4            ;BR IF NOT
13199  052540  000000                          HALT                 ;BREAK-DEPRESS CONTINUE TO CONTINUE
13200  052542  010505                          MOV   SP,R5          ;SAVE THE SP
13201  052544  010737  001010                  MOV   PC,@#$LPERR    ;FOR PROPER SP RESETTING ON ERROR LOOP
13202  052550  010506                    1$:   MOV   R5,SP          ;RESTORE SP FOR ERROR LOOPING
13203  052552  012737  052600  063242          MOV   #3$,@#MBUF1    ;SET UP POINTER--DEST ADDR = 3$ FOR JSR
13204  052560  012746  000020                  MOV   #20,-(SP)      ;SET "T" BIT IN THE NEW PSW
13205  052564  012746  052574                  MOV   #2$,-(SP)      ;MAKE NEW PC = 2$
13206  052570  000257                          CCC                  ;SCOPE SYNC
13207  052572  000006                          RTT                  ;SET "T" BIT - GO TO 2$
```

```
                                USER CONTROLLED BREAKPOINT -- BIT14

13208
13209  052574  004777  010442   2$:     JSR     PC,@MBUF1       ;INSTRUCTION SHOULD SPRING TRAP
13210
13211  052600  104005           3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13212
13213  052602  010506                   MOV     R5,SP           ;RESTORE SP IF ALL OK OR NOT LOOPING
13214                           ;;*****************************************************************
13215                           ;*TEST 717      BUT SERVICE TEST - (JMP A)
13216                           ;;*****************************************************************
13217  052604                   ^ST717:
13218  052604  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13219  052606  012700  000717           MOV     #717,R0         ;;LOAD R0 WITH TEST NUMBER
13220  052612  013701  052632           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13221  052616  012746  000020           MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13222  052622  012746  052632           MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13223  052626  000257                   CCC                     ;SCOPE SYNC
13224  052630  000006                   RTT                     ;SET "T" BIT - GO TO 2$
13225
13226  052632  000167  000000   2$:     JMP     3$              ;JMP INSTRUCTION SHOULD SPRING TRAP
13227
13228  052636  104005           3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13229
13230                           ;;*****************************************************************
13231                           ;*TEST 720      BUT SERVICE TEST - (JMP @A)
13232                           ;;*****************************************************************
13233  052640                   ^ST720:
13234  052640  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13235  052642  012700  000720           MOV     #720,R0         ;;LOAD R0 WITH TEST NUMBER
13236  052646  013701  052674           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13237  052652  012737  052700  063236   MOV     #3$,@#MBUF0     ;SET UP POINTER--DEST ADDR = 3$ FOR JMP
13238  052660  012746  000020           MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13239  052664  012746  052674           MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13240  052670  000257                   CCC                     ;SCOPE SYNC
13241  052672  000006                   RTT                     ;SET "T" BIT - GO TO 2$
13242
13243  052674  000177  010336   2$:     JMP     @MBUF0          ;JMP INSTRUCTION SHOULD SPRING TRAP
13244
13245  052700  104005           3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13246
13247                           ;;*****************************************************************
13248                           ;*TEST 721      BUT SERVICE TEST - (RTS PC)
13249                           ;;*****************************************************************
13250  052702                   ^ST721:
13251  052702  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13252  052704  012700  000721           MOV     #721,R0         ;;LOAD R0 WITH TEST NUMBER
13253  052710  013701  052744           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13254  052714  010605                   MOV     SP,R5           ;SAVE THE SP
13255  052716  010737  001010           MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13256  052722  010506           1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13257  052724  012746  052746           MOV     #3$,-(SP)       ;RTS WILL LOAD PC WITH 3$
13258  052730  012746  000020           MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13259  052734  012746  052744           MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13260  052740  000257                   CCC                     ;SCOPE SYNC
13261  052742  000006                   RTT                     ;SET "T" BIT - GO TO 2$
13262
13263  052744  000207   2$:     RTS     PC              ;RTS INSTRUCTION SHOULD SPRING TRAP
```

```
13264
13265  052746  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13266
13267                                  ;;*****************************************************************
13268                                  ;*TEST 722      ALU ADD FUNCTION TEST
13269                                  ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13270                                  ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13271                                  ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13272
13273                                  ;       AIN     BIN     CIN
13274
13275                                  ;       0       0       0
13276                                  ;       0       0       1
13277                                  ;       0       1       0
13278                                  ;       0       1       1
13279                                  ;       1       0       0
13280                                  ;       1       0       1
13281                                  ;       1       1       0
13282                                  ;       1       1       1
13283
13284                                  ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
13285                                  ;TAGGED "ALUADD" AS SHOWN BELOW:
13286
13287                                          ;ALUADD:        NULL
13288                                  ;                       SRC OP1
13289                                  ;                       DST OP1
13290                                  ;                       SUM1
13291                                  ;                       SRC OP2
13292                                  ;                       DST OP2
13293                                  ;                       SUM2
13294                                  ;                       ETC.
13295
13296                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
13297                                  ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
13298                                  ;;*****************************************************************
13299  052750                          TST722:
13300  052750  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13301  052752  012700  000722                  MOV     #722,R0         ;;LOAD R0 WITH TEST NUMBER
13302  052756  012705  063264          1$:     MOV     #ALUADD+4,R5    ;R5 POINTS TO TABLE OF NO.S
13303  052762  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13304  052766  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
13305                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
13306  052770  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
13307  052772  022705  063342                  CMP     #ALUADD+62,R5   ;DONE ALL NO.S IN TABLE ?
13308  052776  001413                          BEQ     5$              ;BR IF YES
13309  053000  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
13310  053002  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
13311  053004  000257                          CCC                     ;SCOPE SYNC
13312
13313  053006  060103                  2$:     ADD     R1,R3           ;TEST THE ADD FUNCTION
13314
13315  053010  021503                          CMP     (R5),R3         ;CORRECT SUM ?
13316  053012  001766                          BEQ     4$              ;GO ADD NEXT PAIR IF YES
13317
13318  053014  011504                          MOV     (R5),R4         ;GET S / B SUM
13319  053016  014502                          MOV     -(R5),R2        ;GET DEST OP
```

```
13320  053020  104010                  3$:     ERROR   10              ;ALU ADD OPERATION FAILED
13321
13322  053022  005725                          TST     (R5)+           ;CORRECT R5 POINTER
13323  053024  000761                          BR      4$              ;GO DO NEXT PAIR
13324
13325  053026  012737  052756  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13326
13327                                  ;;*****************************************************************
13328                                  ;*TEST 723      ALU SUB FUNCTION TEST
13329                                  ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13330                                  ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13331                                  ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13332
13333                                  ;       AIN     BIN     CIN
13334
13335                                  ;       0       0       0
13336                                  ;       0       0       1
13337                                  ;       0       1       0
13338                                  ;       0       1       1
13339                                  ;       1       0       0
13340                                  ;       1       0       1
13341                                  ;       1       1       0
13342                                  ;       1       1       1
13343
13344                                  ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
13345                                  ;TAGGED "ALUADD" AS SHOWN BELOW:
13346
13347                                          ;ALUSUB:        NULL
13348                                  ;                       SRC OP1
13349                                  ;                       DST OP1
13350                                  ;                       DIFF1
13351                                  ;                       SRC OP2
13352                                  ;                       DST OP2
13353                                  ;                       DIFF2
13354                                  ;                       ETC.
13355
13356                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
13357                                  ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
13358                                  ;;*****************************************************************
13359  053034                          TST723:
13360  053034  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13361  053036  012700  000723                  MOV     #723,R0         ;;LOAD R0 WITH TEST NUMBER
13362  053042  012705  063504          1$:     MOV     #ALUSUB+4,R5    ;R5 POINTS TO TABLE OF NO.S
13363  053046  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13364  053052  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
13365                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
13366  053054  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
13367  053056  022705  063562                  CMP     #ALUSUB+62,R5   ;DONE ALL NO.S IN TABLE ?
13368  053062  001413                          BEQ     5$              ;BR IF YES
13369  053064  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
13370  053066  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
13371  053070  000257                          CCC                     ;SCOPE SYNC
13372
13373  053072  160103                  2$:     SUB     R1,R3           ;TEST THE SUB FUNCTION
13374
13375  053074  021503                          CMP     (R5),R3         ;CORRECT DIFF. ?
```

```
13376  053076  001766                          BEQ     4$              ;GO SUB NEXT PAIR IF YES
13377
13378  053100  011504                          MOV     (R5),R4         ;GET S / B DIFF
13379  053102  014502                          MOV     -(R5),R2        ;GET DEST OP
13380  053104  104010                  3$:     ERROR   10              ;ALU SUB OPERATION FAILED
13381
13382  053106  005725                          TST     (R5)+           ;CORRECT R5 POINTER
13383  053110  000761                          BR      4$              ;GO DO NEXT PAIR
13384
13385  053112  012737  053042  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13386
13387                                  ;;*****************************************************************
13388                                  ;*TEST 724      ALU "AND" FUNCTION TEST USING BIC INSTRUCTION
13389                                  ;THIS TEST VERIFIES THAT THE ALU "AND" FUNCTION RESPONDS CORRECTLY
13390                                  ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
13391                                  ;IT  EXECUTES THE BIC INSTRUCTION FOR THE FOLLOWING PAIRS OF
13392                                  ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
13393
13394                                          ;SOURCE OP      DEST. OP        RESULT
13395
13396                                          ;000000         000000          000000
13397                                          ;177777         177777          000000
13398                                          ;000000         177777          177777
13399                                          ;177777         000000          000000
13400                                          ;125252         125252          000000
13401                                          ;052525         052525          000000
13402                                          ;125252         052525          052525
13403                                          ;052525         125252          125252
13404
13405                                  ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TEBLE TAGGED
13406                                  ;"ANDTAB" IN THE FOLLOWING PATTERN:
13407                                          ;ANDTAB:        NULL
13408                                  ;                       SRC OP1
13409                                  ;                       DST OP1
13410                                  ;                       ANS1
13411                                  ;                       SRC OP2
13412                                  ;                       DST OP2
13413                                  ;                       ANS2
13414                                  ;                       ETC.
13415
13416                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
13417                                  ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
13418                                  ;;*****************************************************************
13419  053120                          TST724:
13420  053120  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13421  053122  012700  000724                  MOV     #724,R0         ;LOAD R0 WITH TEST NUMBER
13422  053126  012705  063344          1$:     MOV     #ANDTAB+4,R5    ;R5 POINTS TO TABLE OF TEST NO.S
13423  053132  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13424  053136  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
13425                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
13426  053140  005725                  4$:     TST     (R5)+           ;POINT TO A SOURCE OPR
13427  053142  022705  063422                  CMP     #ANDTAB+62,R5   ;DONE ALL COMBINATIONS ?
13428  053146  001413                          BEQ     5$              ;BR IF YES
13429  053150  012501                          MOV     (R5)+,R1        ;LOAD THE SRC OP
13430  053152  012503                          MOV     (R5)+,R3        ;LOAD THE DEST OP
13431  053154  000257                          CCC                     ;SCOPE SYNC
```

```
13432
13433  053156  040103                  2$:     BIC     R1,R3           ;TEST THE "AND"
13434
13435  053160  020315                          CMP     R3,(R5)         ;RESULT CORRECT ?
13436  053162  001766                          BEQ     4$              ;BR IF YES - GET THE NEXT PAIR
13437
13438  053164  011504                          MOV     (R5),R4         ;GET THE S / B DATA
13439  053166  014502                          MOV     -(R5),R2        ;GET DEST OP
13440  053170  104010                  3$:     ERROR   10              ;ALU "AND" FAILED
13441
13442  053172  005725                          TST     (R5)+           ;CORRECT R5 POINTER
13443  053174  000761                          BR      4$              ;GO GET NEXT PAIR
13444
13445  053176  012737  053126  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13446
13447                                  ;;*****************************************************************
13448                                  ;*TEST 725       ALU "OR" FUNCTION TEST USING BIS INSTRUCTION
13449                                  ;THIS TEST VERIFIES THAT THE ALU "OR" FUNCTION RESPONDS CORRECTLY
13450                                  ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
13451                                  ;IT  EXECUTES THE BIS INSTRUCTION FOR THE FOLLOWING PAIRS OF
13452                                  ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
13453
13454                                          ;SOURCE OP      DEST. OP        RESULT
13455
13456                                          ;000000         000000          000000
13457                                          ;177777         177777          177777
13458                                          ;000000         177777          177777
13459                                          ;177777         000000          177777
13460                                          ;125252         125252          125252
13461                                          ;052525         052525          052525
13462                                          ;125252         052525          177777
13463                                          ;052525         125252          177777
13464
13465                                  ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TABLE TAGGED
13466                                  ;"ORTAB" IN THE FOLLOWING PATTERN:
13467                                          ;ORTAB: NULL
13468                                  ;                       SRC OP1
13469                                  ;                       DST OP1
13470                                  ;                       ANS1
13471                                  ;                       SRC OP2
13472                                  ;                       DST OP2
13473                                  ;                       ANS2
13474                                  ;                       ETC.
13475
13476                                  ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
13477                                  ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
13478                                  ;;*****************************************************************
13479  053204                          TST725:
13480  053204  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13481  053206  012700  000725                  MOV     #725,R0         ;;LOAD R0 WITH TEST NUMBER
13482  053212  012705  063424          1$:     MOV     #ORTAB+4,R5     ;R5 POINTS TO TABLE OF TEST NO.S
13483  053216  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13484  053222  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
13485                                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
13486  053224  005725                  4$:     TST     (R5)+           ;POINT TO A SOURCE OPR
13487  053226  022705  063502                  CMP     #ORTAB+62,R5    ;DONE ALL COMBINATIONS ?
```

```
13488  053232  001413                        BEQ     5$              ;BR IF YES
13489  053234  012501                        MOV     (R5)+,R1        ;LOAD THE SRC OP
13490  053236  012503                        MOV     (R5)+,R3        ;LOAD THE DEST OP
13491  053240  000257                        CCC                     ;SCOPE SYNC
13492
13493  053242  050103                2$:     BIS     R1,R3           ;TEST THE "OR"
13494
13495  053244  020315                        CMP     R3,(R5)         ;RESULT CORRECT ?
13496  053246  001766                        BEQ     4$              ;BR IF YES - GET THE NEXT PAIR
13497
13498  053250  011504                        MOV     (R5),R4         ;GET THE S / B DATA
13499  053252  014502                        MOV     -(R5),R2        ;GET DEST OP
13500  053254  104010                3$:     ERROR   10              ;ALU "OR" FAILED
13501
13502  053256  005725                        TST     (R5)+           ;CORRECT R5 POINTER
13503  053260  000761                        BR      4$              ;GO GET NEXT PAIR
13504
13505  053262  012737  053212  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13506
13507                                ;;*****************************************************************
13508                                ;*TEST 726       INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
13509                                ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
13510                                ;TEST SEQUENCE:
13511
13512                                        ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
13513                                        ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
13514                                        ;3. THE SOURCE OP IS INCREMENTED
13515                                        ;4. THE DEST OP IS DECREMENTED
13516                                        ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE SOURCE OP GOES
13517                                        ;   NEGATIVE
13518
13519                                ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
13520
13521                                ;       1. SW09=0  THE TEST IS EXITED
13522                                ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
13523
13524                                ;;*****************************************************************
13525  053270                        TST726:
13526  053270  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13527  053272  012700  000726                MOV     #726,R0         ;;LOAD R0 WITH TEST NUMBER
13528  053276  005001                10$:    CLR     R1              ;INITIALIZE REGS TO 000000
13529  053300  005002                        CLR     R2
13530  053302  005004                        CLR     R4
13531  053304  010737  001010                MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13532  053310  010203                1$:     MOV     R2,R3           ;LOAD DEST OPERAND
13533  053312  000257                        CCC                     ;SCOPE SYNC
13534
13535  053314  060103                2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
13536                                                                ;RESULT S / B = 000000
13537
13538  053316  020403                        CMP     R4,R3           ;RESULT = 000000 ?
13539  053320  001402                        BEQ     4$              ;BR IF YES
13540
13541  053322  104010                3$:     ERROR   10              ;INCORRECT RESULT IN R3
13542
13543  053324  000407                        BR      TST727          ;;EXIT TO NEXT TEST
```

```
13544
13545  053326  005201                    4$:     INC     R1              ;ADD 1 TO SOURCE OP
13546  053330  100402                            BMI     5$              ;GET OUT IF IT WENT NEGATIVE
13547  053332  005302                            DEC     R2              ;SUB 1 FROM THE DEST OP
13548  053334  000765                            BR      1$              ;GO ADD THE TWO NO.S
13549
13550  053336  012737  053276  001010    5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
13551
13552                                    ;;*****************************************************************
13553                                    ;*TEST 727      INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
13554                                    ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
13555                                    ;TEST SEQUENCE:
13556
13557                                            ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
13558                                            ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
13559                                            ;3. THE SOURCE OP IS DECREMENTED
13560                                            ;4. THE DEST OP IS INCREMENTED
13561                                            ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE DEST. OP GOES
13562                                            ;    NE    VE
13563
13564                                    ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
13565
13566                                    ;       1. SW09=0  THE TEST IS EXITED
13567                                    ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
13568                                    ;;*****************************************************************
13569  053344                            TST727:
13570  053344  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13571  053346  012700  000727                    MOV     #727,R0         ;;LOAD R0 WITH TEST NUMBER
13572  053352  005001                    10$:    CLR     R1              ;INITIALIZE REGS TO 000000
13573  053354  005002                            CLR     R2
13574  053356  005004                            CLR     R4
13575  053360  010737  001010                    MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
13576  053364  010203                    1$:     MOV     R2,R3           ;LOAD DEST OPERAND
13577  053366  000257                            CCC                     ;SCOPE SYNC
13578
13579  053370  060103                    2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
13580                                            ;RESULT S / B = 000000
13581
13582  053372  020403                            CMP     R4,R3           ;RESULT = 000000 ?
13583  053374  001402                            BEQ     4$              ;BR IF YES
13584
13585  053376  104010                    3$:     ERROR   10              ;INCORRECT RESULT IN R3
13586
13587  053400  000407                            BR      TST730          ;;GO TO SCOPE EXIT
13588
13589  053402  005202                    4$:     INC     R2              ;ADD 1 TO DEST. OP
13590  053404  100402                            BMI     5$              ;GET OUT IF IT WENT NEGATIVE
13591  053406  005301                            DEC     R1              ;SUB 1 FROM THE SOURCE OP
13592  053410  000765                            BR      1$              ;GO ADD THE TWO NO.S
13593
13594  053412  012737  053352  001010    5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
13595
13596                                    ;;*****************************************************************
13597                                    ;*TEST 730      MUL RA,RB TEST ; N:C = 1111
13598                                    ;;*****************************************************************
13599  053420                            TST730:
```

```
13600  053420  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13601  053422  012700  000730                  MOV     #730,R0         ;;LOAD R0 WITH TEST NUMBER
13602  053426  013737  053456  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13603  053434  005001                          CLR     R1              ;S/B RESULT IN R2
13604  053436  012704  000006                  MOV     #6,R4           ;S/B RESULT IN R3
13605  053442  012702  000002                  MOV     #2,R2           ;INITIALIZE REG
13606  053446  005003                          CLR     R3              ;INITIALIZE REG + 1
13607  053450  012705  000003                  MOV     #3,R5           ;INITIALIZE SRC
13608  053454  000277                          SCC                     ;SCOPE SYNC
13609
13610  053456  070205                  2$:     MUL     R5,R2           ;TEST THE MUL
13611
13612  053460  100403                          BMI     3$              ;N:C=0000?
13613  053462  001402                          BEQ     3$
13614  053464  102401                          BVS     3$
13615  053466  103001                          BCC     4$
13616
13617  053470  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13618
13619  053472  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
13620  053474  001002                          BNE     5$              ;BR IF NOT
13621  053476  020102                          CMP     R1,R2           ;REG CORRECT?
13622  053500  001401                          BEQ     TST731          ;;BR IF YES
13623
13624  053502  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13625
13626                                  ;;**************************************************************
13627                                  ;;TEST 731      MUL (RA),RB TEST ; N:C = 0000-SET C
13628                                  ;;**************************************************************
13629  053504                          TST731:
13630  053504  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13631  053506  012700  000731                  MOV     #731,R0         ;;LOAD R0 WITH TEST NUMBER
13632  053512  013737  053546  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13633  053520  005001                          CLR     R1              ;S/B RESULT IN R2
13634  053522  012704  123450                  MOV     #123450,R4      ;S/B RESULT IN R3
13635  053526  012702  012345                  MOV     #012345,R2      ;INITIALIZE REG
13636  053532  005003                          CLR     R3              ;INITIALIZE REG + 1
13637  053534  012705  063236                  MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
13638  053540  012715  000010                  MOV     #10,(R5)        ;INITIALIZE SRC
13639  053544  000257                          CCC                     ;SCOPE SYNC
13640
13641  053546  070215                  2$:     MUL     (R5),R2         ;TEST THE MUL
13642
13643  053550  100403                          BMI     3$              ;N:C=0001?
13644  053552  001402                          BEQ     3$
13645  053554  102401                          BVS     3$
13646  053556  103401                          BCS     4$
13647
13648  053560  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13649
13650  053562  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
13651  053564  001002                          BNE     5$              ;BR IF NOT
13652  053566  020102                          CMP     R1,R2           ;REG CORRECT?
13653  053570  001401                          BEQ     TST732          ;;BR IF YES
13654
13655  053572  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
```

```
13656
13657                                   ;;*********************************************************************
13658                                   ;*TEST 732      MUL (RA)+,RB TEST ; N:C = 0000-SET Z
13659                                   ;;*********************************************************************
13660  053574                           TST732:
13661  053574  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13662  053576  012700  000732                  MOV     #732,R0         ;;LOAD R0 WITH TEST NUMBER
13663  053602  013737  053634  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13664  053610  005001                          CLR     R1              ;S/B RESULT IN R2
13665  053612  005004                          CLR     R4              ;S/B RESULT IN R3
13666  053614  005002                          CLR     R2              ;INITIALIZE REG
13667  053616  012703  177777                  MOV     #-1,R3          ;INITIALIZE REG + 1
13668  053622  012705  063236                  MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
13669  053626  012715  000010                  MOV     #10,(R5)        ;INITIALIZE SRC
13670  053632  000257                          CCC                     ;SCOPE SYNC
13671
13672  053634  070225                  2$:     MUL     (R5)+,R2        ;TEST THE MUL
13673
13674  053636  100403                          BMI     3$              ;N:C=0100?
13675  053640  001002                          BNE     3$
13676  053642  102401                          BVS     3$
13677  053644  103001                          BCC     4$
13678
13679  053646  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13680
13681  053650  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
13682  053652  001002                          BNE     5$              ;BR IF NOT
13683  053654  020102                          CMP     R1,R2           ;REG CORRECT?
13684  053656  001401                          BEQ     6$              ;BR IF YES
13685
13686  053660  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13687
13688  053662  022705  063240          6$:     CMP     #MBUF0+2,R5     ;DID R5 GET AUTO-INCREMENTED?
13689  053666  001401                          BEQ     TST733          ;;BR IF YES
13690
13691  053670  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
13692
13693                                   ;;*********************************************************************
13694                                   ;*TEST 733      MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST = -,+
13695                                   ;;*********************************************************************
13696  053672                           TST733:
13697  053672  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13698  053674  012700  000733                  MOV     #733,R0         ;;LOAD R0 WITH TEST NUMBER
13699  053700  013737  053740  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13700  053706  012701  177777                  MOV     #-1,R1          ;S/B RESULT IN R2
13701  053712  012704  177770                  MOV     #-10,R4         ;S/B RESULT IN R3
13702  053716  012702  000001                  MOV     #1,R2           ;INITIALIZE REG
13703  053722  005003                          CLR     R3              ;INITIALIZE REG + 1
13704  053724  012705  063232                  MOV     #ATA+10,R5      ;SET UP POINTER TO POINTER TO MBUF0
13705  053730  012737  177770  063236          MOV     #-10,@#MBUF0    ;INITIALIZE SRC
13706  053736  000257                          CCC                     ;SCOPE SYNC
13707
13708  053740  070235                  2$:     MUL     @(R5)+,R2       ;TEST THE MUL
13709
13710  053742  100003                          BPL     3$              ;N:C=1000?
13711  053744  001402                          BEQ     3$
```

```
13712  053746  102401                          BVS     3$
13713  053750  103001                          BCC     4$
13714
13715  053752  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13716
13717  053754  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
13718  053756  001002                          BNE     5$              ;BR IF NOT
13719  053760  020102                          CMP     R1,R2           ;REG CORRECT?
13720  053762  001401                          BEQ     6$              ;BR IF YES
13721
13722  053764  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13723
13724  053766  022705  063234          6$:     CMP     #ATA+12,R5      ;DID R5 GET AUTO-INCREMENTED?
13725  053772  001401                          BEQ     TST734          ;;BR IF YES
13726
13727  053774  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
13728
13729                                  ;;**********************************************************************
13730                                  ;*TEST 734      MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK = +,-
13731                                  ;;**********************************************************************
13732  053776                          TST734:
13733  053776  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13734  054000  012700  000734                  MOV     #734,R0         ;;LOAD R0 WITH TEST NUMBER
13735  054004  013737  054044  001076          MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13736  054012  012701  177777                  MOV     #-1,R1          ;S/B RESULT IN R2
13737  054016  012704  177770                  MOV     #-10,R4         ;S/B RESULT IN R3
13738  054022  012702  177777                  MOV     #-1,R2          ;INITIALIZE REG
13739  054026  005003                          CLR     R3              ;INITIALIZE REG + 1
13740  054030  012705  063240                  MOV     #MBUF0+2,R5     ;SET UP POINTER TO SRC
13741  054034  012737  000010  063236          MOV     #10,@#MBUF0     ;INITIALIZE SRC
13742  054042  000277                          SCC                     ;SCOPE SYNC
13743
13744  054044  070245                  2$:     MUL     -(R5),R2        ;TEST THE MUL
13745
13746  054046  100003                          BPL     3$              ;N:C=1000?
13747  054050  001402                          BEQ     3$
13748  054052  102401                          BVS     3$
13749  054054  103001                          BCC     4$
13750
13751  054056  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13752
13753  054060  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
13754  054062  001002                          BNE     5$              ;BR IF NOT
13755  054064  020102                          CMP     R1,R2           ;REG CORRECT?
13756  054066  001401                          BEQ     6$              ;BR IF YES
13757
13758  054070  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13759
13760  054072  022705  063236          6$:     CMP     #MBUF0,R5       ;DID SRC REG GET AUTO-DECREMENTED?
13761  054076  001401                          BEQ     TST735          ;;BR IF YES
13762
13763  054100  104046                          ERROR   46              ;AUTO DECREMENT DID NOT OCCUR
13764
13765                                  ;;**********************************************************************
13766                                  ;*TEST 735      MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
13767                                  ;;**********************************************************************
```

```
13768  054102                          TST735:
13769  054102  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13770  054104  012700  000735          MOV     #735,R0         ;;LOAD R0 WITH TEST NUMBER
13771  054110  013737  054150  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
13772  054116  005001                  CLR     R1              ;S/B RESULT IN R2
13773  054120  012704  106420          MOV     #106420,R4      ;/S/B RESULT IN R3
13774  054124  012702  177776          MOV     #-2,R2          ;INITIALIZE REG
13775  054130  012703  177777          MOV     #-1,R3          ;INITIALIZE REG + 1
13776  054134  012705  063234          MOV     #ATA+12,R5      ;SET UP POINTER TO POINTER TO MBUF0
13777  054140  012737  134570  063236  MOV     #-43210,@#MBUF0 ;INITIALIZE SRC
13778  054146  000277                  SCC                     ;SCOPE SYNC
13779
13780  054150  070255          2$:     MUL     @-(R5),R2       ;TEST THE MUL
13781
13782  054152  100403                  BMI     3$              ;N:C=0001?
13783  054154  001402                  BEQ     3$
13784  054156  102401                  BVS     3$
13785  054160  103401                  BCS     4$
13786
13787  054162  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13788
13789  054164  020304          4$:     CMP     R3,R4           ;REG+1 CORRECT?
13790  054166  001002                  BNE     5$              ;BR IF NOT
13791  054170  020102                  CMP     R1,R2           ;REG CORRECT?
13792  054172  001401                  BEQ     6$              ;BR IF YES
13793
13794  054174  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13795
13796  054176  022705  063232          CMP     #ATA+10,R5      ;DID R5 GET AUTO-DECREMENTED?
13797  054202  001401          6$:     BEQ     TST736          ;;BR IF YES
13798
13799  054204  104046                  ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
13800
13801                          ;;*****************************************************************
13802                          ;*TEST 736       MUL X(RA),RB TEST ; N:C = 1111 TO 0100
13803                          ;;*****************************************************************
13804  0E' 06                  TST736:
13805  054206  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13806  054210  012700  000736          MOV     #736,R0         ;;LOAD R0 WITH TEST NUMBER
13807  054214  013737  054250  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
13808  054222  005001                  CLR     R1              ;S/B RESULT IN R2
13809  054224  005004                  CLR     R4              ;S/B RESULT IN R3
13810  054226  012702  012345          MOV     #012345,R2      ;INITIALIZE REG
13811  054232  012703  177777          MOV     #-1,R3          ;INITIALIZE REG + 1
13812  054236  012705  063236          MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
13813  054242  005065  000002          CLR     2(R5)           ;INITIALIZE SRC
13814  054246  000277                  SCC                     ;SCOPE SYNC
13815
13816  054250  070265  000002  2$:     MUL     2(R5),R2        ;TEST THE MUL
13817
13818  054254  100403                  BMI     3$              ;N:C=0100?
13819  054256  001002                  BNE     3$
13820  054260  102401                  BVS     3$
13821  054262  103001                  BCC     4$
13822
13823  054264  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
```

```
13824
13825  054266  020304                    4$:     CMP     R3,R4           ;REG+1 CORRECT?
13826  054270  001002                            BNE     5$              ;BR IF NOT
13827  054272  020102                            CMP     R1,R2           ;REG CORRECT?
13828  054274  001401                            BEQ     TST737          ;;BR IF YES
13829
13830  054276  104045                    5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13831
13832                                    ;;**********************************************************************
13833                                    ;*TEST 737       MUL @X(RA),RB TEST
13834                                    ;;**********************************************************************
13835  054300                            TST737:
13836  054300  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13837  054302  012700  000737                    MOV     #737,R0         ;;LOAD R0 WITH TEST NUMBER
13838  054306  013737  054344  001076            MOV     @#2$,@#STMP0    ;GET TEST INSTRUCTION WORD
13839  054314  005001                            CLR     R1              ;S/B RESULT IN R2
13840  054316  012704  000100                    MOV     #100,R4         ;S/B RESULT IN R3
13841  054322  012702  000010                    MOV     #10,R2          ;INITIALIZE REG
13842  054326  005003                            CLR     R3              ;INITIALIZE REG + 1
13843  054330  012705  063222                    MOV     #ATA,R5         ;GET POINTER TO TABLE OF POINTERS
13844  054334  012737  000010  063236            MOV     #10,@#MBUF0     ;INITIALIZE SRC
13845  054342  000257                            CCC                     ;SCOPE SYNC
13846
13847  054344  070275  000010           2$:     MUL     @10(R5),R2      ;TEST THE MUL
13848
13849  054350  020304                            CMP     R3,R4           ;REG+1 CORRECT?
13850  054352  001002                            BNE     3$              ;BR IF NOT
13851  054354  020102                            CMP     R1,R2           ;REG CORRECT?
13852  054356  001401                            BEQ     TST740          ;;BR IF YES
13853
13854  054360  104045                    3$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
13855
13856                                    ;;**********************************************************************
13857                                    ;*TEST 740       DIV #N,RA TEST ; N:C = 1111
13858                                    ;;**********************************************************************
13859  054362                            TST740:
13860  054362  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13861  054364  012700  000740                    MOV     #740,R0         ;;LOAD R0 WITH TEST NUMBER
13862  054370  013737  054416  001076            MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
13863  054376  012701  010000                    MOV     #010000,R1      ;S/B RES IN R2
13864  054402  012704  000001                    MOV     #1,R4           ;S/B RES IN R3
13865  054406  005002                            CLR     R2              ;SET UP REG OPERAND
13866  054410  012703  020001                    MOV     #020001,R3      ;SET UP REG+1 OP
13867  054414  000277                            SCC                     ;SCOPE SYNC
13868
13869  054416  071227  000002           2$:     DIV     #2,R2           ;TEST DIV
13870
13871  054422  100403                            BMI     3$              ;N:C=0000?
13872  054424  001402                            BEQ     3$
13873  054426  102401                            BVS     3$
13874  054430  103001                            BCC     4$
13875
13876  054432  104044                    3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13877
13878  054434  020304                    4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
13879  054436  001002                            BNE     5$              ;BR IF NOT
```

```
13880  054440  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
13881  054442  001401                          BEQ     TST741          ;;BR IF YES
13882
13883  054444  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
13884
13885                                  ;;**********************************************************************
13886                                  ;*TEST 741      DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
13887                                  ;;**********************************************************************
13888  054446                          TST741:
13889  054446  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13890  054450  012700  000741                  MOV     #741,R0         ;;LOAD R0 WITH TEST NUMBER
13891  054454  013737  054504  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
13892  054462  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
13893  054466  012704  177776                  MOV     #-2,R4          ;S/B RES IN R3
13894  054472  012702  177777                  MOV     #-1,R2          ;SET UP REG OPERAND
13895  054476  012703  177762                  MOV     #-14.,R3        ;SET UP REG+1 OP
13896  054502  000257                          CCC                     ;SCOPE SYNC
13897
13898  054504  071227  000004          2$:     DIV     #4,R2           ;TEST DIV
13899
13900  054510  100003                          BPL     3$              ;N:C=1000?
13901  054512  001402                          BEQ     3$
13902  054514  102401                          BVS     3$
13903  054516  103001                          BCC     4$
13904
13905  054520  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13906
13907  054522  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
13908  054524  001002                          BNE     5$              ;BR IF NOT
13909  054526  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
13910  054530  001401                          BEQ     TST742          ;;BR IF YES
13911
13912  054532  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
13913
13914                                  ;;**********************************************************************
13915                                  ;*TEST 742      DIV #N,RA TEST ; N:C = 0000 TO 0100
13916                                  ;;**********************************************************************
13917  054534                          TST742:
13918  054534  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13919  054536  012700  000742                  MOV     #742,R0         ;;LOAD R0 WITH TEST NUMBER
13920  054542  013737  054566  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
13921  054550  005001                          CLR     R1              ;S/B RES IN R2
13922  054552  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
13923  054556  005002                          CLR     R2              ;SET UP REG OPERAND
13924  054560  012703  000001                  MOV     #1,R3           ;SET UP REG+1 OP
13925  054564  000257                          CCC                     ;SCOPE SYNC
13926
13927  054566  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
13928
13929  054572  100403                          BMI     3$              ;N:C=0100?
13930  054574  001002                          BNE     3$
13931  054576  102401                          BVS     3$
13932  054600  103001                          BCC     4$
13933
13934  054602  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
13935
```

```
13936  054604  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
13937  054606  001002                          BNE     5$              ;BR IF NOT
13938  054610  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
13939  054612  001401                          BEQ     TST743          ;;BR IF YES
13940
13941  054614  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
13942
13943                                  ;;*****************************************************************
13944                                  ;*TEST 743      DIV #-N,RA TEST ; RA POS
13945                                  ;;*****************************************************************
13946  054616                          TST743:
13947  054616  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13948  054620  012700  000743                  MOV     #743,R0         ;;LOAD R0 WITH TEST NUMBER
13949  054624  013737  054652  001076          MOV     @#2$,@#STMP0    ;GET COPY OF TEST INSTRUCTION
13950  054632  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
13951  054636  012704  000002                  MOV     #2,R4           ;S/B RES IN R3
13952  054642  005002                          CLR     R2              ;SET UP REG OPERAND
13953  054644  012703  000016                  MOV     #14.,R3         ;SET UP REG+1 OP
13954  054650  000257                          CCC                     ;SCOPE SYNC
13955
13956  054652  071227  177774          2$:     DIV     #-4,R2          ;TEST DIV
13957
13958  054656  020304                          CMP     R3,R4           ;CORRECT RESULT IN REG+1?
13959  054660  001002                          BNE     3$              ;BR IF NOT
13960  054662  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
13961  054664  001401                          BEQ     TST744          ;;BR IF YES
13962
13963  054666  104045                  3$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
13964
13965                                  ;;*****************************************************************
13966                                  ;*TEST 744      DIV  TEST - V BIT GETS SET
13967                                  ;*      THIS TEST TESTS THAT THE V BIT CAN BE SET IN ALL THE
13968                                  ;*      POSSIBLE WAYS.  SINCE THE INSTRUCTION SHOULD BE ABORTED, THE
13969                                  ;*      RESULTS CANNOT BE GUARANTEED.  FOR THIS REASON, ONLY
13970                                  ;*      THE CONDITION CODES ARE CHECKED.
13971                                  ;;*****************************************************************
13972  054670                          TST744:
13973  054670  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13974  054672  012700  000744                  MOV     #744,R0         ;;LOAD R0 WITH TEST NUMBER
13975  054676  013701  054716                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13976  054702  012704  000002                  MOV     #2,R4           ;S/B PSW
13977  054706  005037  177776                  CLR     @#PSW           ;CLEAR OUT OTHER PSW BITS
13978  054712  012702  000050                  MOV     #50,R2          ;SET UP REG OP
13979
13980  054716  071227  000005          2$:     DIV     #5,R2           ;TEST DIV -- SHOULD ABORT
13981
13982  054722  100424                          BMI     3$              ;N:C=0010?
13983  054724  001423                          BEQ     3$
13984  054726  102022                          BVC     3$
13985  054730  103421                          BCS     3$
13986
13987  054732  012702  177777                  MOV     #-1,R2          ;INITIALIZE REG OP
13988  054736  005003                          CLR     R3              ;INITILAIZE REG+1 OP
13989
13990  054740  071227  177776                  DIV     #-2,R2          ;TEST DIV -- SHOULD ABORT
13991
```

```
13992  054744  100413                    BMI     3$              ;N:C=0010?
13993  054746  001412                    BEQ     3$
13994  054750  102011                    BVC     3$
13995  054752  103410                    BCS     3$
13996
13997  054754  012704  000003            MOV     #3,R4           ;S/B PSW
13998
13999  054760  071227  000000            DIV     #0,R2           ;TEST DIV BY 0 -- SHOULD ABORT
14000
14001  054764  100403                    BMI     3$              ;N:C=0010?
14002  054766  001402                    BEQ     3$
14003  054770  102001                    BVC     3$
14004  054772  103405                    BCS     TST745          ;;IF ALL OK, THEN EXIT TEST
14005
14006  054774  013703  177776    3$:     MOV     @#PSW,R3        ;GET WAS PSW
14007  055000  012702  177776            MOV     #PSW,R2         ;DESTINATION IS PSW
14008
14009  055004  104001                    ERROR   1               ;CONDITION CODES SET WRONG
14010
14011                              ;;*****************************************************************
14012                              ;*TEST 745      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14013                              ;;*****************************************************************
14014  055006                      TST745:
14015  055006  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14016  055010  012700  000745            MOV     #745,R0         ;;LOAD R0 WITH TEST NUMBER
14017  055014  013701  055032            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14018  055020  012704  123450            MOV     #123450,R4      ;S/B RESULT
14019  055024  012703  112345            MOV     #112345,R3      ;INITIAL REG
14020  055030  000257                    CCC                     ;SCOPE SYNC
14021
14022  055032  072327  000003    2$:     ASH     #3,R3           ;TEST THE ASH
14023
14024  055036  100003                    BPL     3$              ;N:C=1010?
14025  055040  001402                    BEQ     3$
14026  055042  102001                    BVC     3$
14027  055044  103001                    BCC     4$
14028
14029  055046  104002    3$:     ERROR   2               ;INCORRECT CONDITION CODES
14030
14031  055050  020304    4$:     CMP     R3,R4           ;CORRECT RESULT?
14032  055052  001401                    BEQ     TST746          ;;BR IF YES
14033  055054  104002                    ERROR   2               ;ASH DELIVERED WRONG RESULT
14034
14035                              ;;*****************************************************************
14036                              ;*TEST 746      ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14037                              ;;*****************************************************************
14038  055056                      TST746:
14039  055056  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14040  055060  012700  000746            MOV     #746,R0         ;;LOAD R0 WITH TEST NUMBER
14041  055064  013701  055102            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14042  055070  005004                    CLR     R4              ;S/B RESULT
14043  055072  012703  000004            MOV     #4,R3           ;INITIAL REG
14044  055076  000257                    CCC                     ;SCOPE SYNC
14045  055100  000270                    SEN                     ;CODES = 1000
14046
14047  055102  072327  177775    2$:     ASH     #-3,R3          ;TEST THE ASH
```

```
14048
14049  055106  100403                          BMI     3$              ;N:C=0101?
14050  055110  001002                          BNE     3$
14051  055112  102401                          BVS     3$
14052  055114  103401                          BCS     4$
14053
14054  055116  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14055
14056  055120  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14057  055122  001401                          BEQ     TST747          ;;BR IF YES
14058  055124  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14059
14060                                  ;;*****************************************************************
14061                                  ;*TEST 747       ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14062                                  ;;*****************************************************************
14063  055126                          TST747:
14064  055126  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14065  055130  012700  000747                  MOV     #747,R0         ;;LOAD R0 WITH TEST NUMBER
14066  055134  013701  055152                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14067  055140  012704  177234                  MOV     #177234,R4      ;S/B RESULT
14068  055144  012703  123432                  MOV     #123432,R3      ;INITIAL REG
14069  055150  000277                          SCC                     ;SCOPE SYNC
14070
14071  055152  072327  177772          2$:     ASH     #-6,R3          ;TEST THE ASH
14072
14073  055156  100003                          BPL     3$              ;N:C=1000?
14074  055160  001402                          BEQ     3$
14075  055162  102401                          BVS     3$
14076  055164  103001                          BCC     4$
14077
14078  055166  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14079
14080  055170  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14081  055172  001401                          BEQ     TST750          ;;BR IF YES
14082  055174  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14083
14084                                  ;;*****************************************************************
14085                                  ;*TEST 750       ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14086                                  ;;*****************************************************************
14087  055176                          TST750:
14088  055176  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14089  055200  012700  000750                  MOV     #750,R0         ;;LOAD R0 WITH TEST NUMBER
14090  055204  013737  055234  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14091  055212  012701  123456                  MOV     #123456,R1      ;S/B RES IN R2
14092  055216  012704  076530                  MOV     #076530,R4      ;S/B RES IN R3
14093  055222  012702  112345                  MOV     #112345,R2      ;INITIALIZE COMBINED
14094  055226  012703  147653                  MOV     #147653,R3      ;  REGISTERS
14095  055232  000257                          CCC                     ;SCOPE SYNC
14096
14097  055234  073227  000003          2$:     ASHC    #3,R2           ;TEST ASHC
14098
14099  055240  100003                          BPL     3$              ;N:C=1010?
14100  055242  001402                          BEQ     3$
14101  055244  102001                          BVC     3$
14102  055246  103001                          BCC     4$
14103
```

```
14104  055250  104044                  3$:     ERROR   44              ;COND CODES WRONG
14105
14106  055252  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14107  055254  001002                          BNE     5$              ;BR IF NOT
14108  055256  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14109  055260  001401                          BEQ     TST751          ;;BR IF YES
14110  055262  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14111
14112                                  ;;**********************************************************************
14113                                  ;*TEST 751      ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14114                                  ;;**********************************************************************
14115  055264                          TST751:
14116  055264  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14117  055266  012700  000751                  MOV     #751,R0         ;;LOAD R0 WITH TEST NUMBER
14118  055272  013737  055316  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14119  055300  005001                          CLR     R1              ;S/B RES IN R2
14120  055302  005004                          CLR     R4              ;S/B RES IN R3
14121  055304  005002                          CLR     R2              ;INITIALIZE COMBINED
14122  055306  012703  000005                  MOV     #5,R3           ;  REGISTERS
14123  055312  000257                          CCC                     ;SCOPE SYNC
14124  055314  000270                          SEN                     ;CODES = 1000
14125
14126  055316  073227  177775          2$:     ASHC    #-3,R2          ;TEST ASHC
14127
14128  055322  100403                          BMI     3$              ;N:C=0101?
14129  055324  001002                          BNE     3$
14130  055326  102401                          BVS     3$
14131  055330  103401                          BCS     4$
14132
14133  055332  104044                  3$:     ERROR   44              ;COND CODES WRONG
14134
14135  055334  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14136  055336  001002                          BNE     5$              ;BR IF NOT
14137  055340  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14138  055342  001401                          BEQ     TST752          ;;BR IF YES
14139  055344  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14140
14141                                  ;;**********************************************************************
14142                                  ;*TEST 752      ASHC #N,RA TEST ; SHIFT RIGHT ; N:C = 1111 TO 1000
14143                                  ;;**********************************************************************
14144  055346                          TST752:
14145  055346  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14146  055350  012700  000752                  MOV     #752,R0         ;;LOAD R0 WITH TEST NUMBER
14147                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT15
14148  055354  032737  100000  063160          BIT     #BIT15,@#BPTLOC ;BREAKPOINT HALT SET ??
14149  055362  001401                          BEQ     .+4             ;BR IF NOT
14150  055364  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
14151  055366  013737  055416  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14152  055374  012701  177234                  MOV     #177234,R1      ;S/B RES IN R2
14153  055400  012704  135275                  MOV     #135275,R4      ;S/B RES IN R3
14154  055404  012702  123456                  MOV     #123456,R2      ;INITIALIZE COMBINED
14155  055410  012703  127542                  MOV     #127542,R3      ;  REGISTERS
14156  055414  000257                          CCC                     ;SCOPE SYNC
14157
14158  055416  073227  177772          2$:     ASHC    #-6,R2          ;TEST ASHC
14159
```

```
14160  055422  100003                  BPL     3$              ;N:C=1000?
14161  055424  001402                  BEQ     3$
14162  055426  102401                  BVS     3$
14163  055430  103401                  BCS     4$
14164
14165  055432  104044          3$:     ERROR   44              ;COND CODES WRONG
14166
14167  055434  020102          4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14168  055436  001002                  BNE     5$              ;BR IF NOT
14169  055440  020403                  CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14170  055442  001401                  BEQ     TST753          ;;BR IF YES
14171  055444  104045          5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14172
14173                          ;*
14174                          ;*      THIS SECTION OF THE MED TESTS EXERCISES CERTAIN SCRATCH
14175                          ;*      PAD REGISTERS USING MED READS AND WRITES.  THEIR ORIGINAL
14176                          ;*      CONTENTS ARE RESTORED BUT:
14177                          ;*
14178                          ;*      ******************** IMPORTANT NOTE ********************
14179                          ;*      *                                                      *
14180                          ;*      *  THE CONSOLE MUST NOT !!! BE USED DURING THESE MED   *
14181                          ;*      *  TESTS. NO INTERRUPTS OR TRAPS CAN BE ALLOWED EITHER *
14182                          ;*      *                                                      *
14183                          ;*      ********************************************************
14184                          ;*
14185
14186                          ;;*********************************************************************
14187                          ;*TEST 753      CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
14188                          ;*      THE NEXT TEST BELOW CHECKS TO SEE THAT THE "MED"
14189                          ;*      (MAINTENANCE, EXAM, AND DEPOSIT) INSTRUCTION WILL EXECUTE
14190                          ;*      WHEN IN KERNEL MODE WITHOUT AFFECTING THE PSW AND
14191                          ;*      THAT IT IS ILLEGAL IN USER MODE
14192                          ;;*********************************************************************
14193  055446                  TST753:
14194  055446  012700  000752          MOV     #752,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14195  055452  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14196  055454  012737  000304  177770  MED1:   MOV     #304,@#UBREAK   ;SET SCOPE SYNC FOR MED INSTR
14197  055462  012737  140000  177776          MOV     #140000,@#PSW   ;GO TO USER MODE
14198  055470  012706  001000          MOV     #STACK,SP       ;SETUP USER STACK PTR.
14199  055474  012737  055526  000004          MOV     #2$,@#ERRVEC    ;SET ERROR TRAP VECTOR TO 2$ BELOW
14200  055502  012737  055526  000010          MOV     #2$,@#RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
14201  055510  012701  177777          MOV     #-1,R1          ;LOAD R1 WITH A -1
14202  055514  005000                  CLR     R0              ;CLEAR R0
14203  055516  076600                  MED                     ;TRY TO DO MAINT. EXAMINE
14204  055520  000041                  .WORD   041             ;MED READ CODE FOR R1
14205  055522  104012                  ERROR   12              ;ERROR - MED INST. NOT ILLEGAL IN USER
14206  055524  000404                  BR      4$
14207  055526  005700          2$:     TST     R0              ;IS R0 UNCHANGED?
14208  055530  001401                  BEQ     3$              ;BRANCH IF YES
14209  055532  104013                  ERROR   13              ;ERROR - MED INSTRUCTION WAS EXECUTED
14210                                                          ;BEFORE TRAPPING
14211  055534  022626          3$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14212  055536  012737  061144  000004  4$:     MOV     #BERR,@#ERRVEC  ;RESTORE ERROR TRAP VECTOR
14213  055544  012737  061046  000010          MOV     #RSERR,@#RESVEC ;RESTORE RESERVED INST. TRAP VECTOR
14214
14215  055552  005037  177776  MEDO:   CLR     @#PSW           ;GO TO KERNEL MODE,CLEAR COND. CODES
```

```
14216  055556  076600                          MED                     ;DO MAINT. EXAMINE OF R1
14217  055560  000041                          .WORD   041             ;MED READ CODE FOR R1
14218  055562  103403                          BCS     MEDHLT
14219  055564  102402                          BVS     MEDHLT
14220  055566  100401                          BMI     MEDHLT
14221  055570  001001                          BNE     .+4
14222  055572  104014                  MEDHLT: ERROR   14              ;ERROR CC-BITS IN PSW AFFECTED BY MED
14223
14224                                  ;;**********************************************************************
14225                                  ;*TEST 754      MED TEST - R/W DATA PATTERNS TO REGS
14226                                  ;*      THIS PARTICULAR MED TEST WRITES DATA PATTERNS
14227                                  ;*      TO THOSE INTERNAL REGS. WHICH CAN BE WRITTEN
14228                                  ;*      AND READ WITHOUT SPECIAL CONSIDERAIONS.  REGISTERS
14229                                  ;*      REQUIRING SPECIAL TESTS ARE TESTED IN LATER
14230                                  ;*      MED TESTS.
14231                                  ;*      TABLE II CONTAINS THE REGISTER ADDRESSES.
14232                                  ;*
14233                                  ;*      A MAX. OF 3 ERRORS ARE REPORTED FOR EACH LOC.
14234                                  ;*
14235                                  ;;**********************************************************************
14236  055574                          TST754:
14237  055574  012700  000753                 MOV     #753,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14238  055600  000004                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14239  055602  012737  000340  177776 MEDT1:  MOV     #340,@#PSW      ;KERNEL MODE-PRIORITY 7
14240  055610  012701  064112                 MOV     #TBL2,R1        ;INITIALIZE ADDRESS POINTER
14241  055614  012737  125252  001102 1$:     MOV     #125252,@#$TMP2
14242  055622  111137  055670                 MOVB    (R1),@#11$      ;PUT WRITE CODE BY "WRITE-MED'S"
14243  055626  112137  055712                 MOVB    (R1)+,@#13$     ;AND POINT R1 TO READ CODE
14244  055632  111137  055650                 MOVB    (R1),@#10$      ;PUT READ CODE BY "READ-MED'S"
14245  055636  112137  055676                 MOVB    (R1)+,@#12$     ;R1 NOW POINTS TO NEXT REG.
14246  055642  005037  001106                 CLR     @#$TMP4         ;CLEAR ERROR COUNTER
14247  055646  076600                  2$:    MED                     ;MED-READ THE INTERNAL REG.
14248  055650  000000                  10$:   .WORD   0               ;MED-READ CODE
14249  055652  010037  001076                 MOV     R0,@#$TMP0      ;SAVE ITS ORIGINAL CONTENTS
14250  055656  010137  001100                 MOV     R1,@#$TMP1      ;SAVE ADDR. PTR. VALUE
14251  055662  013700  001102                 MOV     @#$TMP2,R0      ;LOAD R0 WITH DATA TO BE WRITTEN
14252  055666  076600                  MED                     ;MED-WRITE THE TEST DATA
14253  055670  000000                  11$:   .WORD   0               ;MED-WRITE CODE
14254  055672  005000                  CLR     R0                      ;CLEAR R0
14255  055674  076600                  MED                     ;MED-READ THE DATA BACK
14256  055676  000000                  12$:   .WORD   0               ;MED-READ CODE
14257  055700  010037  001104                 MOV     R0,@#$TMP3      ;SAVE DATA READ FOR COMPARISON
14258  055704  013700  001076                 MOV     @#$TMP0,R0      ;LOAD ORIGINAL DATA IN R0
14259  055710  076600                  MED                     ;MED-WRITE ORG. DATA TO REG.
14260  055712  000000                  13$:   .WORD   0               ;MED-WRITE CODE
14261  055714  023737  001102  001104         CMP     @#$TMP2,@#$TMP3 ;DID DATA READ=DATA WRITTEN?
14262  055722  001412                         BEQ     3$              ;BRANCH IF YES
14263  055724  013737  055676  001100         MOV     @#12$,@#$TMP1   ;SAVE MED-CODE FOR ERROR
14264  055732  022737  000003  001106         CMP     #3,@#$TMP4      ;MAX. ERROR REPORTS YET?
14265  055740  002401                         BLT     14$             ;BRANCH IF YES
14266  055742  104022                         ERROR   22              ;INT. REG. READ BACK WRONG DATA
14267  055744  005237  001106          14$:   INC     @#$TMP4         ;INCREMENT ERROR COUNTER
14268  055750  005137  001102          3$:    COM     @#$TMP2         ;CHANGE DATA PATTERN
14269  055754  013701  001100                 MOV     @#$TMP1,R1      ;RESTORE ADDR. POINTER
14270  055760  022737  125252  001102         CMP     #125252,@#$TMP2 ;BOTH DATA PATTERNS BEEN USED?
14271  055766  001327                         BNE     2$              ;BRANCH IF NO
```

```
14272  055770  005711                          TST     (R1)            ;END OF ADDR. TABLE?
14273  055772  001310                          BNE     1$              ;BRANCH IF NO
14274
14275                                  ;;*****************************************************************
14276                                  ;*TEST 755      MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14277                                  ;*
14278                                  ;*      THIS TEST CHECKS ALL OF THE "NOP" OPERATION CODES
14279                                  ;*      TO ENSURE THEY WILL EXECUTE AS NOP'S AND
14280                                  ;*      NOT RESULT IN A PROCESSOR HANG.  THE "NOPS"
14281                                  ;*      TABLE (TABLE III) HOLDS THESE CODES.
14282                                  ;*      THIS TEST ALSO READS THE PROGRAM COUNTER (R7) VALUES
14283                                  ;*      STORED IN A & B SCRATCH PADS TO SEE THAT THEY
14284                                  ;*      READ PROPERLY.  THE R7 ADDRESSES ARE IN TABLE IV.
14285                                  ;*
14286                                  ;;*****************************************************************
14287  055774
14288  055774  012700  000754                  MOV     #754,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14289  056000  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14290  056002  012701  064330          MEDT3:  MOV     #TBL3,R1        ;INITIALIZE NOP TABLE PTR. (R1)
14291  056006  112137  056014          1$:     MOVB    (R1)+,@#10$     ;PLACE FIRST "NOP-CODE" AFTER MED
14292                                                                  ;AND POINT R1 TO LAST CODE IN GROUP
14293  056012  076600                  5$:     MED                     ;EXECUTE MED WITH NOP OP-CODE
14294  056014  000000                  10$:    .WORD   0
14295  056016  123711  056014                  CMPB    @#10$,(R1)      ;HAVE ALL NOPS IN THAT GROUP
14296                                                                  ;BEEN TESTED?
14297  056022  103003                          BHIS    6$              ;BRANCH IF YES
14298  056024  005237  056014                  INC     @#10$           ;NEXT NOP IN GROUP
14299  056030  000770                          BR      5$
14300  056032  105721                  6$:     TSTB    (R1)+           ;POINT R1 TO NEXT NOP GROUP
14301  056034  005711                          TST     (R1)            ;HAVE ALL GROUPS BEEN TESTED
14302  056036  001363                          BNE     1$              ;BRANCH IF NO
14303
14304  056040  113737  064351  056052  MEDT4:  MOVB    @#R7A+1,@#5$    ;LOAD R7A READ CODE AFTER MED
14305  056046  005000                  4$:     CLR     R0              ;CLEAR R0
14306  056050  076600                          MED                     ;MED READ R7 IN THE ASP
14307  056052  000000                  5$:     .WORD   0               ;READ CODE FOR R7A
14308  056054  020027  056054                  CMP     R0,#5$+2        ;DID R7A READ CORRECTLY?
14309  056060  001411                          BEQ     6$              ;BRANCH IF YES
14310  056062  013737  056052  001100          MOV     @#5$,@#$TMP1    ;SAVE MED-CODE FOR ERROR
14311  056070  012737  056054  001102          MOV     #5$+2,@#$TMP2   ;SAVE DATA EXPECTED
14312  056076  010037  001104                  MOV     R0,@#$TMP3      ;SAVE DATA RECEIVED
14313  056102  104022                          ERROR   22              ;R7A DID NOT READ THE RIGHT VALUE
14314  056104  023727  056052  000047  6$:     CMP     @#5$,#47        ;HAS R7B BEEN CHECKED?
14315  056112  001404                          BEQ     8$              ;BRANCH IF YES
14316  056114  113737  064355  056052          MOVB    @#R7B+1,@#5$    ;LOAD R7B READ CODE AFTER MED
14317  056122  000751                          BR      4$              ;TEST R7 BSP
14318  056124                          8$:
14319
14320
14321                                  ;;*****************************************************************
14322                                  ;*TEST 756      MED TEST - CSP CONSTANTS CHECK
14323                                  ;*
14324                                  ;*      THIS TEST CHECKS THE CONSTANT VALUES LOCATED
14325                                  ;*      IN THE C SCRATCH PAD.  THE CONSTANTS ARE READ
14326                                  ;*      WITH A MED INSTRUCTION AND COMPARED TO THEIR
14327                                  ;*      EXPECTED VALUE.  THE ADDRESSES OF THESE CONSTANTS
```

```
14328                                   ;*      AND THE VALUES EXPECTED ARE IN TABLE VII.
14329                                   ;*
14330                                   ;;*****************************************************************
14331  056124                           TST756:
14332  056124  012700  000755                  MOV     #755,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14333  056130  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14334
14335  056132  170000                  MEDT10: CFCC                    ;EXECUTE FLT. PT INST. SO FLT. PT.
14336                                                                  ;CONSTANTS ARE LOADED INTO CSP
14337  056134  012701  064462                  MOV     #TBL7,R1        ;SETUP TABLE POINTER
14338  056140  012167  000006          10$:    MOV     (R1)+,1$        ;LOAD MED READ CODE AT 1$
14339  056144  001414                          BEQ     11$             ;BR IF END OF TABLE
14340  056146  005000                          CLR     R0
14341  056150  076600                          MED                     ;READ INTERNAL CONTENTS INTO R0
14342  056152  000000                  1$:     .WORD   0
14343  056154  020021                          CMP     R0,(R1)+        ;WAS THE CONSTANT READ THE ONE EXPECTED
14344  056156  001770                          BEQ     10$             ;BRANCH IF YES
14345  056160  013737  056152  001100          MOV     @#1$,@#$TMP1    ;SAVE MEDCODE FOR ERROR
14346  056166  016137  177776  001102          MOV     -2(R1),@#$TMP2  ;SAVE CONSTANT VALUE EXPECTED
14347  056174  104021                          ERROR   21              ;CSP LOCATION HELD WRONG VALUE
14348  056176                          11$:
14349
14350
14351                                   ;;*****************************************************************
14352                                   ;*TEST 757      MED TEST - MICROBK CHECK OF MICRO-POINTS
14353                                   ;*
14354                                   ;*      THIS TEST USES THE MICROBREAK REGISTER AND THE
14355                                   ;*      INFORMATION IN TABLE V TO CHECK THAT THE
14356                                   ;*      CORRECT MED-FLOW IS ENTERED WHEN EACH
14357                                   ;*      REGISTER IS ACCESSED BY A MED INSTRUCTION.
14358                                   ;*      THE MICROBREAK REG. IS SETUP TO CAUSE A TRAP TO
14359                                   ;*      LOC. 4 WHEN ITS CONTENTS EQUAL THE ADDRESS
14360                                   ;*      OF THE MIRCOWORD BEING EXECUTED.
14361                                   ;*
14362                                   ;*          NOTE:  THE MICRO BREAK - TRAP-TO-4 CAPABILITY
14363                                   ;*                 IS TRIED AT THE BEGINNING OF THE TEST.
14364                                   ;*                 IF IT DOESN'T WORK, AN ERROR IS PRINTED
14365                                   ;*                 AND THE TEST IS SKIPPED
14366                                   ;*
14367                                   ;*****************************************************************
14368  056176                           TST757:
14369  056176  012700  000756                  MOV     #756,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14370  056202  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14371  056204  012737  000071  177770  MEDT11: MOV     #SWB01,@#UBREAK ;LOAD MICROBK. REG. WITH AN MICRO ADDR.
14372  056212  012737  061030  000004          MOV     #BKROUT,@#4     ;LOAD ADDR. OF MICROBK. ROUTINE IN 4
14373  056220  012737  000340  000006          MOV     #340,@#6        ;LOAD KERNEL PSW - PRIORITY 7 IN 6
14374  056226  005037  061036                  CLR     @#BKFLAG        ;CLEAR MICROBK. TRAP FLAG
14375  056232  076600                          MED                     ;GET WHAMI INTO R0
14376  056234  000022                          RDWHAMI
14377  056236  052700  001000                  BIS     #BIT9,R0        ;SET BIT 9
14378  056242  076600                          MED                     ;MED-WRITE THE WHAMI REG TO
14379  056244  000222                  10$:    WRWHAMI                 ;ENABLE MICROBK-TRAP-TO-4
14380  056246  076600                          MED                     ;GET FLAG REGISTER
14381  056250  000144                          RDFLAG
14382  056252  052700  100000                  BIS     #BIT15,R0       ;SET BIT 15 IN R0
14383  056256  076600                          MED                     ;MED-WRITE THE FLAG REG TO
```

```
14384  056260  000344                  11$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
14385  056262  000300                          SWAB    R0              ;MICROBK TRAP SHOULD OCCUR ON SWAB
14386  056264  005737  061036                  TST     @#BKFLAG        ;DID TRAP TO 4 OCCUR?
14387  056270  001007                          BNE     1$              ;BRANCH IF YES
14388  056272  005037  001076                  CLR     @#STMP0
14389  056276  016737  121567  001100          MOV     SWB01,@#STMP1   ;SAVE EXPECTED UBREAK ADDR
14390  056304  104015                          ERROR   15              ;MICROBREAK TRAP DIDN'T WORK
14391  056306  000453                          BR      50$             ;SKIP TO END OF TEST
14392
14393  056310  012701  000710          1$:     MOV     #SWB01*10,R1    ;GET CORRECT U-ADDR
14394  056314  076600                          MED                     ;GET LOG CUA REG
14395  056316  000103                          RDLCUA
14396  056320  042700  100007                  BIC     #100007,R0      ;GET RID OF IRRELEVANT BITS
14397  056324  020001                          CMP     R0,R1           ;WAS CORRECT UADDR LOGGED?
14398  056326  001401                          BEQ     3$              ;BR IF YES
14399  056330  104025                          ERROR   25              ;CUA CONTAINS INCORRECT U-ADDR
14400  056332  012701  064362          3$:     MOV     #TBL5,R1                ;INTIALIZE TABLE PTR. (R1)
14401  056336  012702  064410                  MOV     #TBL6,R2
14402  056342  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP RETURN TO 2$
14403  056346  111137  056404          2$:     MOVB    (R1),@#12$      ;LOAD WRITE CODE AFTER MED
14404  056352  001431                          BEQ     50$             ;BR IF END OF TABLE
14405  056354  011237  177770          4$:     MOV     (R2),@#UBREAK   ;LOAD MICROBK REG. WITH MICROADDR.
14406  056360  005037  061036                  CLR     @#BKFLAG        ;CLEAR MICROBK TRAP-TO-4 FLAG
14407  056364  076600                          MED                     ;GET FLAG REGISTER
14408  056366  000144                          RDFLAG
14409  056370  052700  100000                  BIS     #BIT15,R0       ;SET BIT 15 IN R0
14410  056374  076600                          MED                     ;MED WRITE TO FLAG REG TO
14411  056376  000344                  15$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
14412  056400  005000                          CLR     R0              ;IN CASE U-BREAK TRAP DOESN'T OCCOR
14413                                                                   ;USUALLY BETTER TO WRITE 0'S
14414  056402  076600                          MED
14415  056404  000000                  12$:    .WORD   0
14416  056406  005737  061036                  TST     @#BKFLAG        ;DID WE TRAP-TO-4? (FLAG NOT = 0)
14417  056412  001006                          BNE     20$             ;BRANCH IF YES TO NEXT ENTRY
14418  056414  013737  056404  001076          MOV     @#12$,@#STMP0   ;SAVE MED-CODE FOR ERROR
14419  056422  011237  001100                  MOV     (R2),@#STMP1    ;SAVE EXPECTED U-ADDR FOR ERROR
14420  056426  104015                          ERROR   15              ;MICROBK. TRAP-TO-4 DID NOT OCCUR
14421
14422  056430  105721                  20$:    TSTB    (R1)+           ;INCREMENT TO NEXT TABLE
14423  056432  005722                          TST     (R2)+           ;ENTRIES AND
14424  056434  000744                          BR      2$              ;CONTINUE
14425
14426  056436  076600                  50$:    MED                     ;GET WHAMI INTO R0
14427  056440  000022                          RDWHAMI
14428  056442  042700  001000                  BIC     #BIT9,R0        ; CLEAR BIT 9
14429  056446  076600                          MED                     ;CLEAR THE FLAG REG. TO
14430  056450  000344                  13$:    WRFLAG                  ;    DISABLE MICROBK. TRAPPING
14431  056452  076600                          MED                     ;CLEAR THE WHAMI REG. TO
14432  056454  000222                  14$:    WRWHAMI                 ;    DISABLE MICROBK. TRAP-TO-4
14433  056456  012737  056204  001010          MOV     #MEDT11,@#$LPERR        ;RESET LOOP ON ERROR POINTER
14434  056464  012737  061144  000004          MOV     #BERR,@#4       ;RESTORE NORMAL ERROR ROUTINE
14435  056472  012737  000304  177770          MOV     #304,@#UBREAK   ;GENERATE SYNC PULSE ON MED INSTR
14436
14437
14438                                  ;;*****************************************************************
14439                                  ;*TEST 760       PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING
```

```
14440                                   ;*      THIS TEST CHECKS THAT THE PROPER PHYSICAL ADDRESS BITS
14441                                   ;*      <17:00> ARE LOGGED UPON ERROR.  THE ERROR IS CAUSED BY
14442                                   ;*      FORCING AN ODD ADDRESS TRAP.  THE ERROR LOG MODE USED
14443                                   ;*      IS "LOG FIRST".  ALSO, THE ODD ADDRESS ERROR BITS IN
14444                                   ;*      THE LOG JAM AND CPU ERROR REGISTER ARE CHECKED.
14445                                   ;;*****************************************************************
14446  056500                           ;ST760:
14447  056500  012700  000757                  MOV     #757,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14448  056504  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14449  056506  012737  056546  000004  1$:     MOV     #2$,@#4         ;SETUP PC FOR ODD ADDR SERVICE
14450  056514  012737  000340  000006          MOV     #340,@#6
14451  056522  012700  100001                  MOV     #BIT15+BIT0,R0  ;SETUP "LOG FIRST" MODE
14452  056526  076600                          MED
14453  056530  000222                          WRWHAMI
14454  056532  012702  056507                  MOV     #1$+1,R2        ;SAVE ADDRESS OF ODD ADDR. INSTRUCTION
14455  056536  005767  177745                  TST     1$+1            ;DO ODD ADDRESS INSTRUCTION TO FORCE
14456                                                                   ;A JAMUPP & TRAP TO 4
14457  056542  104023                          ERROR   23              ;*** ODD ADDR. TRAP DID NOT OCCUR
14458  056544  000441                          BR      10$             ;EXIT TEST
14459  056546  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14460  056550  012737  061144  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW
14461  056556  076600                          MED
14462  056560  000100                          RDLJAM
14463  056562  013701  177766                  MOV     @#CPUERR,R1
14464  056566  032701  000100                  BIT     #BIT6,R1          ;WAS ODD ADDR. ERROR RECORDED BY
14465                                                                   ;THE CPU ERROR REGISTER?
14466  056572  001001                          BNE     3$              ;BRANCH IF YES
14467  056574  104024                          ERROR   24              ;*** CPU ERROR REG. DID NOT
14468                                                                   ;REPORT ODD ADDRESS ERROR
14469                                                                   ;READ THE LOG JAM REGISTER
14470  056576  032700  100004          3$:     BIT     #BIT15+BIT2,R0  ;WAS ODD ADDR. ERROR LOGGED BY LOG JAM
14471  056602  001001                          BNE     4$              ;BRANCH IF YES
14472  056604  104024                          ERROR   24              ;*** LOG JAM REG.  DID NOT LOG
14473                                                                   ;ODD ADDRESS ERROR CORRECTLY
14474
14475  056606  005005                  4$:     CLR     R5              ;CLR ERROR FLAG
14476  056610  076600                          MED                     ;READ THE LOG PBA REGISTER
14477  056612  000102                          RDLPBA
14478  056614  010003                          MOV     R0,R3           ;SAVE RECEIVED PHYS ADDR <15:0>
14479  056616  020002                          CMP     R0,R2           ;WERE BITS <15:00> OF THE PHYSICAL
14480                                                                   ;BUS ADDR. LOGGED CORRECTLY?
14481  056620  001401                          BEQ     5$              ;BRANCH IF YES
14482  056622  005205                          INC     R5              ;SET ERROR FLAG
14483  056624  076600                  5$:     MED                     ;READ THE LOG SERVICE REGISTER
14484  056626  000101                          RDLSERVICE
14485  056630  000300                          SWAB    R0              ;GET "PBA 17&16" DOWN TO BIT POSITION 0&1
14486  056632  042700  177774                  BIC     #177774,R0
14487  056636  001002                          BNE     11$             ;BR IF PHYS ADDR BITS <17:16> LOGGED CORRECTLY
14488  056640  005705                          TST     R5              ;PREVIOUS ERROR?
14489  056642  001402                          BEQ     10$             ;BR IF NOT
14490  056644  005001                  11$:    CLR     R1              ;SET UP EXPECTED PA<17:16>
14491  056646  104026                          ERROR   26              ;*** PHYSICAL BUS ADDR. <17:00>
14492                                                                   ;NOT LOGGED CORRECTLY WHEN
14493                                                                   ;ODD ADDRESS TRAP OCCURRED
14494  056650  005000                  10$:    CLR     R0
14495  056652  076600                          MED                     ;DISABLE "LOG FIRST" MODE
```

```
14496  056654  000222                          WRWHAMI
14497
14498
14499                                  ;;*****************************************************************
14500                                  ;*TEST 761      CHECK DISABLE PARITY ERROR TRAP
14501                                          ;*THIS TEST CHECKS THAT PARITY ERROR TRAPS TO LOCATION 114
14502                                          ;*ARE DISABLED WHEN BIT0 OF THE CACHE CONTROL REGISTER IS
14503                                          ;*SET (=1).  A TRAP TO 114 SHOULD NOT OCCUR AND ERROR
14504                                          ;*INFORMATION SHOULD NOT BE LOGGED IN THE LOG PBA, LOG
14505                                          ;*CACHE DATA, OR LOG TAG DATA REGISTERS.  WRONG PARITY IS
14506                                          ;*WRITTEN INTO A TEST LOCATION TO CAUSE THE PARITY ERROR
14507                                          ;*NEEDED IN THIS TEST.
14508                                  ;;*****************************************************************
14509  056656                          TST761:
14510  056656  012700  000760                 MOV     #760,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14511  056662  000004                 SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14512
14513  056664  012701  063772                 MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
14514  056670  005711                 TST     (R1)                    ;MAKE IT A HIT
14515  056672  012737  000100  177746         MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
14516  056700  012711  125252                 MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
14517  056704  012737  000001  177746         MOV     #DPTRP,@#CCR    ;DISABLE PARITY ERROR TRAPS
14518                                                                  ;AND CLEAR WWP
14519  056712  012737  056752  000114         MOV     #1$,@#114       ;SETUP PARITY ERROR VECTOR
14520  056720  012737  000340  000116         MOV     #340,@#116
14521  056726  005000                 CLR     R0
14522  056730  076600                 MED                             ;CLEAR LOG PBA REGISTER
14523  056732  000302                 WRLPBA
14524  056734  076600                 MED                             ;CLEAR LOG CACHE DATA REGISTER
14525  056736  000306                 WRLDATA
14526  056740  076600                 MED                             ;CLEAR LOG CACHE TAG REGISTER
14527  056742  000307                 WRLTAG
14528  056744  005767  005022                 TST     TLOC1           ;READ TEST LOC0 TO FORCE PARITY ERROR
14529  056750  000406                 BR      2$                      ;BRANCH IF NO TRAP OCCURS
14530  056752  012700  000200         1$:     MOV     #200,R0
14531  056756  076600                 MED                             ;CLEAN UP THE CACHE
14532  056760  000352                 352                             ;INITIALIZATION CODE
14533  056762  022626                 CMP     (SP)+,(SP)+             ;CLEAN UP STACK
14534  056764  104030                 ERROR   30                      ;*** PARITY TRAP TO 114 OCCURRED
14535                                                                  ;WHEN IT SHOULD HAVE BEEN DISABLED
14536  056766  012700  000200         2$:     MOV     #200,R0
14537  056772  076600                 MED                             ;CLEAN UP THE CACHE
14538  056774  000352                 352                             ;INITIALIZATION CODE
14539  056776  012711  125252                 MOV     #125252,(R1)    ;WRITE BAK GOOD PARITY IN TST LOC
14540  057002  012737  000116  000114         MOV     #116,@#114      ;RESTORE ORIGINAL PARITY HANDLER & PSW
14541  057010  005037  000116                 CLR     @#116
14542  057014  005005                 CLR     R5                      ;CLEAR ERROR FLAG
14543  057016  076600                 MED                             ;READ LOG PBA REGISTER
14544  057020  000102                 RDLPBA
14545  057022  010003                 MOV     R0,R3                   ;SAVE COPY
14546                                                                  ;LOG PBA REG. STILL CLEAR?
14547  057024  001401                 BEQ     3$                      ;BRANCH IF YES
14548  057026  005205                 INC     R5                      ;OTHERWISE SET ERROR FLAG
14549  057030  076600         3$:     MED                             ;READ LOG CACHE DATA REG.
14550  057032  000106                 RDLDATA
14551  057034  010001                 MOV     R0,R1                   ;SAVE COPY
```

```
14552
14553  057036  001401                          BEQ     4$              ;LOG CACHE DATA REG. STILL CLEAR?
14554  057040  005205                          INC     R5              ;BRANCH IF YES
14555  057042  076600                  4$:     MFD                     ;OTHERWISE SET ERROR FLAG
14556  057044  000107                          RDLTAG                  ;READ LOG CACHE TAG REG.
14557  057046  010002                          MOV     R0,R2           ;SAVE COPY
14558                                                                  ;LOG CACHE TAG REG. STILL CLEAR?
14559  057050  001401                          BEQ     5$              ;BRANCH IF YES
14560  057052  005205                          INC     R5              ;OTHERWISE SET ERROR FLAG
14561  057054  005705                  5$:     TST     R5              ;WERE ANY OF LOG REGISTERS CHANGED
14562  057056  001401                          BEQ     6$              ;BRANCH IF NO
14563  057060  104027                          ERROR   27              ;*** ONE OF LOG REGISTERS CHANGED
14564                                                                  ;WHEN ERROR SHOULD NOT HAVE BEEN LOGGED
14565                                                                  ;LOG PBA, LOG DATA & LOG TAG
14566                                                                  ;REGISTER SHOULD BE CLEAR.
14567  057062  005037  177746          6$:     CLR     @#CCR           ;ENABLE PARITY ERROR TRAPS
14568
14569                                  ;;**********************************************************************
14570                                  ;*TEST 762       CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
14571
14572                                          ;*THIS TEST CHECKS THAT ALL OF THE PARITY ERROR BITS (5,6,7)
14573                                          ;*OF THE MEMORY ERROR REGISTER ARE SET TO "1" WHEN A CACHE
14574                                          ;*PARITY ERROR OCCURS IN THE BACKUP MODE.
14575                                  ;;**********************************************************************
14576  057066                          TST762:
14577  057066  012700  000761                  MOV     #761,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14578  057072  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14579  057074  012701  063772                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
14580  057100  005711                          TST     (R1)            ;MAKE IT A HIT
14581  057102  012737  000100  177746          MOV     #HWP,@#CCR      ;SET WRITE WRONG PARITY BIT
14582  057110  012711  125252                  MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
14583  057114  042737  000100  177746          BIC     #HWP,@#CCR      ;CLEAR HWP
14584  057122  012737  057156  000114          MOV     #1$,@#114       ;SETUP NEW TEST HANDLER AT PARITY VECTOR
14585  057130  012737  000340  000116          MOV     #340,@#116
14586  057136  005737  063772                  TST     @#TLOC1         ;READ TEST LOC. TO FORCE PARITY ERROR
14587  057142  012700  000200                  MOV     #200,R0
14588  057146  076600                          MFD                     ;CLEAN UP THE CACHE
14589  057150  000352                          352                     ;INITIALIZATION CODE
14590  057152  104031                          ERROR   31              ;*** PARITY ERROR DID NOT CAUSE TRAP
14591  057154  000405                          BR      2$              ;BRANCH TO 2$
14592  057156  012700  000200          1$:     MOV     #200,R0
14593  057162  076600                          MFD                     ;CLEAN UP THE CACHE
14594  057164  000352                          352                     ;INITIALIZATION CODE
14595  057166  022626                          CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14596  057170  022737  000340  177744  2$:     CMP     #000340,@#MEMERR  ;WERE PARITY ERROR BITS (5,6,7) SET
14597                                                                  ;AND CPU ABORT BIT (15) LEFT CLEAR
14598                                                                  ;IN MEMORY ERROR REGISTER?
14599  057176  001403                          BEQ     3$              ;BRANCH IF YES
14600  057200  013700  177744                  MOV     @#MEMERR,R0
14601  057204  104032                          ERROR   32              ;*** MEMORY ERROR REGISTER BITS
14602                                                                  ;WERE SET INCORRECTLY
14603  057206  012737  000116  000114  3$:     MOV     #116,@#114      ;RESTORE OLD PARITY HANDLER PC & PSW
14604  057214  005037  000116                  CLR     @#116
14605
14606                                  ;;**********************************************************************
14607                                  ;*TEST 763       CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
```

```
14608
14609                                   ;*THIS TEST CHECKS THAT THE "UNIBUS TIMEOUT" BIT (BIT4)
14610                                   ;*GETS SET IN THE CPU ERROR REGISTER WHEN A TIMEOUT OCCURS.
14611                                   ;*A TIMEOUT TRAP IS FORCED BY REFERENCING BUS ADDRESS 760000.
14612                                   ;*THEN AN ODD ADDRESS ERROR IS FORCED AND IT
14613                                   ;*IS CHECKED IF ONLY BIT (6)-ODD ADDRESS ERROR IS SET
14614                                   ;*(IN CPUERR).  THIS CHECKS THAT THE ERROR LOG IS
14615                                   ;*CONTINUOSLY UPDATED IN THE "LOG CONTINUOUS" MODE.
14616                                   ;;*****************************************************************
14617  057220                           ;TST763:
14618  057220  012700  000762          MOV     #762,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14619  057224  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14620  057226  012737  057250  000004  MOV     #1$,@#4         ;SETUP NEW PC & PSW FOR THE
14621  057234  012737  000340  000006  MOV     #340,@#6        ;TIMEOUT SERVICE ROUTINE
14622  057242  005737  160000          TST     @#160000        ;FORCE A TIMEOUT TRAP TO 4 BY
14623                                                          ;REFERENCING NON-EXISTENT ADDRESS
14624  057246  000461                  BR      6$
14625  057250  022626          1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14626  057252  012737  061144  000004  MOV     #BERR,@#4       ;RESTORE OLD PC & PSW FOR TIMEOUT
14627  057260  076600                  MED
14628  057262  000100                  RDLJAM
14629  057264  013701  177766          MOV     @#CPUERR,R1     ;SAVE CPU ERR REG
14630  057270  022701  000020          CMP     #BIT4,R1        ;DID "UNIBUS TIMEOUT" BIT IN CPU ERROR
14631                                                          ;REGISTER GET SET?
14632  057274  001401                  BEQ     2$              ;BRANCH IF YES
14633  057276  104033                  ERROR   33              ;*** "UNIBUS TIMEOUT" BIT (BIT4) IN CPU
14634                                                          ;ERROR REG. DID NOT SET WHEN A
14635                                                          ;TIMEOUT WAS FORCED
14636                                                          ;READ THE LOG JAM REGISTER
14637  057300  022700  021200  2$:     CMP     #BIT12+BIT9+BIT7,R0     ;DID "UNIBUS TIMEOUT" BIT (BIT7) SET?
14638                                                          ;BIT 9= POWER STATUS, ALWAYS SET
14639  057304  001401                  BEQ     3$              ;BRANCH IF YES
14640  057306  104033                  ERROR   33              ;*** "UNIBUS TIMEOUT" BIT (BIT7)
14641                                                          ;DID NOT SET IN LOG JAM REGISTER
14642                                                          ;WHEN UNIBUS TIMEOUT WAS FORCED
14643  057310  076600          3$:     MED                     ;READ LOG PBA
14644  057312  000102                  RDLPBA
14645  057314  020027  160000          CMP     R0,#160000      ;WAS PHYS BA LOGGED CORRECTLY?
14646  057320  001403                  BEQ     5$
14647  057322  012701  160000          MOV     #160000,R1
14648  057326  104020                  ERROR   20              ;PHYSICAL BUS ADDRESS WAS
14649                                                          ;LOGGED WRONG ON A UNIBUS
14650                                                          ;TIMEOUT
14651  057330  012737  057352  000004  5$:     MOV     #4$,@#4         ;SET UP PC,PSW FOR ODD ADDRESS
14652  057336  012737  000340  000006  MOV     #340,@#6
14653  057344  005767  177741          TST     3$+1            ;FORCE ODD ADDRESS ERROR
14654  057350  000420                  BR      6$
14655  057352  022626          4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14656  057354  012737  061144  000004  MOV     #BERR,@#4
14657  057362  076600                  MED
14658  057364  000100                  RDLJAM
14659  057366  013701  177766          MOV     @#CPUERR,R1
14660  057372  022701  000100          CMP     #BIT6,R1        ;ODD ADDR. BUT SET 3
14661  057376  001401                  BEQ     7$
14662  057400  104024                  ERROR   24              ;ODD ADDRESS BIT WAS
14663                                                          ;NOT SET IN THE CPU
```

```
14664                                                           ;ERROR REGISTER.  IN LOG
14665                                                           ;CONTINUOUS MODE THE
14666                                                           ;LATEST ERROR SHOULD
14667                                                           ;BE LOGGED
14668  057402  032700  000004          7$:     BIT     #BIT2,R0        ;ODD ADR. BIT SET IN
14669  057406  001001                          BNE     6$              ;LOG JAM?
14670  057410  104024                          ERROR   24              ;ODD ADDRESS BIT WAS
14671                                                           ;NOT SET IN THE LOG
14672                                                           ;JAM REGISTER ON A
14673                                                           ;ODD ADDRESS ERROR
14674  057412  076600                  6$:     MED                     ;CHECK IF LAST INTERRUPT VECTOR
14675  057414  000104                          RDLFGINT                ;WAS LOGGED?
14676  057416  120027  000004                  CMPB    R0,#4
14677  057422  001401                          BEQ     8$
14678  057424  104036                          ERROR   36              ;LAST ERROR VECTOPR WS NOT LOGGED
14679
14680  057426                          8$:
14681
14682
14683                                   ;;*****************************************************************
14684                                   ;*TEST 764      CHECK ILLEGAL INTERNAL ADDRESS TRAP
14685
14686                                           ;*THIS TEST CHECKS THAT A TRAP OCCURS UPON REFERENCING AN
14687                                           ;*ILLEGAL INTERNAL ADDRESS AND THAT "ILLEGAL INTERNAL ADDRESS"
14688                                           ;*BIT (BIT0) OF THE CPU ERROR REGISTER AND BITS OF LOG JAM
14689                                           ;*REGISTER GET SET.  IT ALSO CHECKS IF THE INTERRUPT VECTOR
14690                                           ;*(4) IS SAVED AS THE "LAST INTERRUPT VECTOR" IN THE LOG
14691                                           ;*FLAG/INTERRUPT REG.
14692                                   ;;*****************************************************************
14693  057426                          TST764:
14694  057426  012700  000763                  MOV     #763,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14695  057432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14696  057434  012737  057464  000004          MOV     #1$,@#4         ;SETUP NEW HANDLER PC & PSW
14697  057442  012737  000340  000006          MOV     #340,@#6
14698  057450  005037  177746                  CLR     @#CCR
14699  057454  012707  177746                  MOV     #CCR,PC         ;ILLEGAL INTERNAL ADDRESS TRAP SHOULD OCCUR
14700  057460  104034                          ERROR   34              ;*** ILLEGAL INTERNAL ADDRESS
14701                                                           ;DID NOT RESULT IN A TRAP
14702  057462  000420                          BR      3$              ;BRANCH TO EXIT IF NO TRAP
14703  057464  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14704  057466  012737  061144  000004          MOV     #BERR,@#4       ;RESTORE OLD HANDLER PC & PSW
14705  057474  076600                          MED
14706  057476  000100                          RDLJAM
14707  057500  013701  177766                  MOV     @#CPUERR,R1
14708  057504  032701  000001                  BIT     #BIT0,R1        ;DID "ILLEGAL INTERNAL ADDRESS" BIT (0)
14709                                                           ;IN CPU ERROR REGISTER GET SET?
14710  057510  001001                          BNE     2$              ;BRANCH IF YES
14711  057512  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS
14712                                                           ;BIT DID NOT SET IN CPU ERROR REG.
14713                                                           ;READ THE LOG JAM REG.
14714  057514  032700  000040          2$:     BIT     #BIT5,R0        ;DID "ILLEGAL INTERNAL ADDRESS" BIT (5)
14715                                                           ;IN LOG JAM REG. GET SET
14716  057520  001001                          BNE     3$              ;BRANCH IF YES
14717  057522  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS BIT
14718                                                           ;DID NOT SET IN LOG JAM REG.
14719  057524                          3$:
```

```
14720
14721
14722                                   ;;*****************************************************************
14723                                   ;*TEST 765      CHECK LOG SERVICE & MEMERR LOGS LO-HI BYTE & TAG, IN CACHE ABORT MODE
14724                                           ;TEST CHECKS THAT "LO BYTE PARITY" "HI BYTE PARITY" AND "TAG PARITY"
14725                                           ;BITS CAN SET IN "LOG SERVICE" REGISTERS.  IT IS ALSO
14726                                           ;CHECKED THAT THE PROPER TAG AND DATA BITS GET STORED
14727                                           ;IN THE "LOG CACHE DATA, "LOG CACHE TAG/CPU" AND THE
14728                                           ;"MEMORY ADDRESS REGISTER" WHEN A PARITY ERROR IS
14729                                           ;FORCED.
14730                                           ;IT IS CHECKED IF THE INSTRUCTION WAS ABORTED AND THE
14731                                           ;LOG FLAG/INTERRUPT REGISTER LOGGED THE LAST INTERRUPT
14732                                           ;VECTOR.
14733                                   ;;*****************************************************************
14734  057524                           ↑ST765:
14735  057524  012700  000764                  MOV     #764,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14736  057530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14737
14738  057532  012737  000201  177745          MOV     #DPTRP+PABORT,@#CCR  ;DISABLE PARITY TRAPS (CACHE)
14739  057540  005037  001062                  CLR     @#$REGO
14740  057544  012701  063772                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOC.
14741  057550  005711                          TST     (R1)            ;MAKE IT A HIT
14742  057552  052737  000100  177746          BIS     #WWP,@#CCR      ;WRITE WRONG PARITY SET
14743  057560  012711  125252                  MOV     #125252,(R1)    ;WRITE TEST LOCATION WITH WRONG PARITY
14744  057564  042737  000100  177746          BIC     #WWP,@#CCR      ;CLEAR WWP
14745  057572  012700  100001                  MOV     #BIT15+BIT0,R0
14746  057576  076600                          MED                     ;ENABLE "LOG FIRST" MODE, AND
14747  057600  000222                          WRWHAMI                 ;ERROR LOGGING
14748  057602  042737  000001  177746          BIC     #DPTRP,@#CCR    ;ENABLE CACHE PARITY TRAPS
14749  057610  012737  057636  000114          MOV     #PTRP1,@#114    ;NEW PARITY TRAP SERVICE
14750  057616  016737  004150  001062          MOV     TLOC1,@#$REGO   ;READ TEST LOC, FORCE PARITY ERROR
14751  057624  012700  000200                  MOV     #200,R0
14752  057630  076600                          MED                     ;CLEAN UP THE CACHE
14753  057632  000352                          352                     ;INITIALIZATION CODE
14754  057634  104031                          ERROR   31              ;*** CACHE PARITY ERROR TRAP
14755                                                                  ;DID NOT OCCUR WHEN
14756                                                                  ;TEST LOC WITH BAD PARITY
14757                                                                  ;WAS READ
14758
14759                                                                  ;ENTER HERE IF PARITY TRAP OCCURRED
14760  057636  012700  000200          PTRP1:  MOV     #200,R0
14761  057642  076600                          MED                     ;CLEAN UP THE CACHE
14762  057644  000352                          352                     ;INITIALIZATION CODE
14763  057646  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE CACHE PARITY ERROR TRAPS
14764  057654  012737  000116  000114          MOV     #116,@#114      ;REESTABLISH OLD SERVICE VECTORS
14765  057662  005037  000116                  CLR     @#116
14766  057666  022626                          CMP     (SP)+,(SP)+
14767  057670  005737  001062                  TST     @#$REGO         ;WAS THE INSTRUCTION ABORTED ON
14768                                                                  ;CACHE PARITY ERROR (ABORT MODE)?
14769  057674  001401                          BEQ     1$              ;YES
14770  057676  104041                          ERROR   41              ;INSTRUCTION HAVING CACHE PARITY
14771                                                                  ;ERROR WAS NOT ABORTED, IN THE
14772                                                                  ;CACHE ABORT MODE.
14773  057700  076600                  1$:     MED                     ;READ THE "LOG SERVICE" REGISTER
14774  057702  000101                          RDLSERVICE
14775  057704  010004                          MOV     R0,R4           ;COPY
```

```
14776  057706  042704  177435          BIC     #^C<LO+HI+TAG+BIT1>,R4  ;MASK ALL BUT LO,HI,TAG BITS
14777  057712  022704  000342          CMP     #342,R4                 ;LO,HI ,TAG, CACHE PARITY BITS SET? IN "SERVICE"
14778  057716  001401                  BEQ     2$                      ;YES
14779  057720  104042                  ERROR   42                      ;*** "LO BYTE" PARITY ERROR
14780                                                                  ;AND "TAG" PARITY ERROR BITS
14781                                                                  ;WERE NOT LOGGED CORRECTLY IN "LOG
14782                                                                  ;SERVICE" REGISTER, WHEN PARITY
14783                                                                  ;ERROR TRAP WAS FORCED.
14784                                                                  ;CLEAR BITS ARE ACTIVE.
14785  057722  013700  177744  2$:     MOV     @#MEMERR,R0             ;GET MEM ERR REG
14786  057726  022700  100340          CMP     #HI+LO+TAG+BIT15,R0     ;DID"LO BYTE" "HI BYTE" AND "TAG"
14787                                                                  ;PARITY ERROR BITS SET IN
14788                                                                  ;THE MEMORY ERROR REGISTER?
14789  057732  001401                  BEQ     3$                      ;YES
14790  057734  104043                  ERROR   43                      ;*** "LO BYTE" "HI BYTE" AND "TAG" PARITY
14791                                                                  ;ERROR BITS DID NOT SET
14792                                                                  ;CORRECTLY IN THE MEMORY
14793                                                                  ;ERROR REGISTER
14794  057736  076600          3$:     MED                             ;READ "LOG PBA" REGISTER
14795  057740  000102                  RDLPBA
14796  057742  020027  063772          CMP     R0,#TLOC1               ;DID "LOG PBA" CONTA N CORRECT
14797                                                                  ;PHYSICAL BUS ADDRES .-WHERE
14798                                                                  ;THE PARITY ERROR OC URRED?
14799  057746  001403                  BEQ     4$                      ;YES
14800  057750  012701  063772          MOV     #TLOC1,R1               ;EXPECTED PBA
14801  057754  104020                  ERROR   20                      ;*** PHYSICAL BUS ADDRESS
14802                                                                  ;(WHERE PARITY ERROR OCCURRED)
14803                                                                  ;WAS NOT LOGGED CORRECTLY
14804                                                                  ;WHEN CACHE PARITY ERROR WAS FORCED
14805  057756  076600          4$:     MED
14806  057760  000107                  RDLTAG                          ;READ "LOG CACHE TAG" REGISTER
14807  057762  000300                  SWAB    R0
14808  057764  012701  063772          MOV     #TLOC1,R1               ;SHIFT RIGHT (3 TIMES) THE 16 BIT
14809  057770  000301                  SWAB    R1
14810  057772  106201                  ASRB    R1                      ;PHYSICAL BUS ADDRESS OF THE
14811  057774  106201                  ASRB    R1                      ;TEST LOCATION
14812  057776  106201                  ASRB    R1
14813  060000  052701  000200          BIS     #BIT7,R1                ;FUDGE TAGE BIT
14814  060004  120100                  CMPB    R1,R0                   ;WAS THE CORRECT TAG LOGGED?
14815  060006  001401                  BEQ     5$                      ;YES
14816  060010  104017                  ERROR   17                      ;TAG BITS WERE NOT LOGGED
14817                                                                  ;CORRECTLY, WHEN CACHE
14818                                                                  ;PARITY ERROR WAS FORCED
14819  060012  076600          5$:     MED                             ;READ CACHE DATA
14820  060014  000106                  RDLDATA
14821  060016  020027  125252          CMP     R0,#125252              ;CACHE DATA LOGGED CORRECTLY?
14822  060022  001403                  BEQ     6$
14823  060024  012701  125252          MOV     #125252,R1              ;EXPECTED DATA
14824  060030  104016                  ERROR   16
14825
14826  060032  012700  000001  6$:     MOV     #BIT0,R0                ;SET UP LOG CONTINUOUS
14827  060036  076600                  MED
14828  060040  000222                  WRWHAMI
14829  060042  012737  060054  000004  MOV     #7$,@#4                 ;SETUP CPU VECTOR
14830  060050  005737  160000          TST     @#160000                ;FORCE TIMEOUT & TRAP TO 7$
14831  060054  022626          7$:     CMP     (SP)+,(SP)+
```

```
14832  060056  012737  061144  000004          MOV     #BERR,@#4       ;RESTORE CPU VECTOR
14833  060064  076600                          MED                     ;READ LOG FLAG/INTERRUPT REGISTER
14834  060066  000104                          RDLFGINT
14835  060070  120027  000114                  CMPB    R0,#114         ;DID LO BYTE CONTAIN VECTOR 114?
14836  060074  001403                          BEQ     8$
14837  060076  010037  001062                  MOV     R0,@#$REG0
14838  060102  104036                          ERROR   36              ;LAST INTERRUPT VECTOR WAS NOT
14839                                                                  ;LOGGED CORRECTLY IN FLAG REGISTER
14840                                                                  ;WHEN A CACHE PARITY ERROR WAS
14841                                                                  ;FORCED.
14842  060104                          8$:
14843
14844                                  ;;*****************************************************************
14845                                  ;*TEST 766      CHECK "LOG FIRST" MODE OF ERROR LOGGING
14846                                          ;*THIS TEST CHECKS THE "LOG FIRST" MODE OF ERROR LOGGING.
14847                                          ;*THE "LOG FIRST" MODE IS ENABLED.  THEN A TIME-OUT TRAP
14848                                          ;*IS FORCED, BIT 4 OF CPU ERROR REGISTER SHOULD BE SET.
14849                                          ;*THEN AN ODD ADDRESS TRAP IS FORCED.  HOWEVER, THIS
14850                                          ;*TIME THE ERROR SHOULD NOT BE LOGGED; BIT 6 (ODD
14851                                          ;*ADDRESS) SHOULD NOT BE SET BECAUSE THE ERROR LOG
14852                                          ;*IS LOCKED UP AFTER THE FIRST ERROR.
14853
14854                                          ;*THEN, THE ERROR LOG IS ENABLED (BY SETTING BIT 0 OF
14855                                          ;*WHAMI).  AN ODD ADDRESS ERROR IS FORCED AGAIN AND IT IS
14856                                          ;*CHECKED THAT THIS TIME THE ERROR IS LOGGED, (BIT 6-ODD
14857                                          ;*ADDRESS SHOULD BE SET IN CPU ERROR REGISTER).
14858                                  ;;*****************************************************************
14859  060104                          TST766:
14860  060104  012700  000765                  MOV     #765,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14861  060110  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14862
14863  060112  012700  100001                  MOV     #BIT15+BIT0,R0  ;SET UP "LOG FIRST MODE
14864  060116  076600                          MED
14865  060120  000222                          WRWHAMI
14866  060122  012737  060144  000004          MOV     #1$,@#4         ;SET UP NEW PC & PSW FOR
14867  060130  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT
14868  060136  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT
14869  060142  000462                          BR      5$              ;SKIP TEST IF NO TIMEOUT
14870
14871  060144  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14872                                                                  ;BIT 4 OF CPU ERROR REGISTER
14873                                                                  ;SHOULD HAVE SET
14874  060146  012737  060162  000004          MOV     #2$,@#4         ;SET UP NEW PC FOR ODD ADDRESS
14875  060154  005767  177765                  TST     1$+1            ;FORCE ODD ADDRESS TRAP
14876  060160  000453                          BR      5$              ;SKIP TEST IF NO ODD ADDRESS TRAP
14877
14878  060162  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
14879  060164  012737  061144  000004          MOV     #BERR,@#4
14880  060172  076600                          MED
14881  060174  000100                          RDLJAM
14882  060176  013701  177766                  MOV     @#CPUERR,R1
14883  060202  022701  000020                  CMP     #BIT4,R1        ;"TIMEOUT" BIT SHOULD BE STILL
14884                                                                  ;SET, CHECK?
14885  060206  001402                          BEQ     3$
14886  060210  104033                          ERROR   33              ;*** SECOND ERROR (ODD ADDRESS)
14887                                                                  ;UPDATED THE ERROR LOG IN
```

```
14888                                                                   ;THE LOG FIRST MODE.  BIT 4
14889                                                                   ;(UNIBUS TIMEOUT) SHOULD BE
14890                                                                   ;STILL SET FROM THE FIRST
14891                                                                   ;ERROR
14892  060212  000436                      BR      5$                  ;SKIP THE REST
14893  060214  032700  100004      3$:     BIT     #BIT2+BIT15,R0      ;CHECK THAT ODD ADRES ERROR BITS NOT
14894  060220  001401                      BEQ     6$                  ;SET IN LOG JAM.  NOTE LOG FIRST
14895                                                                   ;MODE SHOULD INHIBIT FURTHER
14896                                                                   ;ERROR LOGGING
14897  060222  104037                      ERROR   37                  ;ODD ADRESS ERROR BITS GOT SET IN LOG JAM
14898                                                                   ;THEY SHOULD NOT BE SINCE LOG FIRST MODE
14899                                                                   ;INHIBITS ERROR LOGGING AFTER THE FIRST ERROR
14900  060224  012700  100001      6$:     MOV     #BIT15+BIT0,R0      ;ENABLE ERROR LOG AGAIN IN
14901                                                                   ;LOG FIRST MODE
14902  060230  076600                      MED
14903  060232  000222                      WRWHAMI
14904  060234  012737  060256  000004      MOV     #4$,@#4             ;SET UP NEW PC & PSW FOR
14905  060242  012737  000340  000006      MOV     #340,@#6            ;ODD ADDRESS ERROR
14906  060250  005767  177741              TST     3$+1                ;FORCE ODD ADDRESS TRAP
14907  060254  000415                      BR      5$                  ;SKIP IF NO TRAP
14908  060256  022626              4$:     CMP     (SP)+,(SP)+         ;RESTORE STACK
14909                                                                   ;RESTORE OLD PC(4), PSW(6)
14910  060260  012737  061144  000004      MOV     #BERR,@#4
14911  060266  022737  000100  177766      CMP     #BIT6,@#CPUERR      ;THE ERROR LOG FROM PREVIOUS
14912                                                                   ;ERROR SHOULD BE OVER WRITTEN.
14913                                                                   ;ODD ADDRESS BIT SHOULD
14914                                                                   ;BE SET, BECAUSE THE ERROR
14915  060274  001405                      BEQ     5$                  ;LOG WAS ENABLED.
14916                                                                   ;OK, IF YES
14917  060276  076600                      MED
14918  060300  000100                      RDLJAM
14919  060302  013701  177766              MOV     @#CPUERR,R1
14920  060306  104040                      ERROR   40                  ;THE ERROR LOG WAS NOT UPDATED
14921                                                                   ;(UPON AN ODD ADDRESS ERROR)
14922                                                                   ;AFTER THE LOG WAS ENABLED.
14923                                                                   ;AT THIS FORMAT BIT 6 OF
14924                                                                   ;CPU ERROR REGISTER SHOULD
14925                                                                   ;BE SET.  IT WAS NOT.
14926  060310  012737  061144  000004  5$:     MOV     #BERR,@#4           ;RESTORE OLD PC(4), PSW(6)
14927  060316  012700  000001              MOV     #BIT0,R0
14928  060322  076600                      MED                         ;PUT THE LOGGING BACK INTO
14929  060324  000222                      WRWHAMI                     ;"CONTINUOUS" MODE
14930
14931                                  ;;*****************************************************************
14932                                  ;*TEST 767      CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
14933                                  ;;*****************************************************************
14934  060326                          TST767:
14935  060326  012700  000766              MOV     #766,R0             ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14936  060332  000004                      SCOPE                       ;CALL THE SCOPE LOOP UTILITY
14937
14938  060334  012737  060344  000030      MOV     #1$,@#30            ;LOAD EMT VECTOR WITH 1$
14939  060342  104000                      EMT                         ;FIRST INTERRUPT -- EMT
14940  060344  022626              1$:     CMP     (SP)+,(SP)+         ;CLEAN UP STACK
14941  060346  012737  061544  000030      MOV     #$ERROR,@#30        ;RESTORE VECTOR
14942  060354  012737  060366  000004      MOV     #2$,@#4             ;SET UP CPU VECTOR
14943  060362  005737  160000              TST     @#160000            ;FORCE TIMEOUT
```

```
14944  060366  022626                    2$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14945  060370  012737  061144  000004            MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
14946  060376  076600                            MED                     ;CHECK FLAG
14947  060400  000104                            RDLFGINT
14948  060402  120027  000030                    CMPB    R0,#30          ;EMT VECTOR LAST LOGGED?
14949  060406  001401                            BEQ     3$              ;BR IF YES
14950  060410  104036                            ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
14951                                                                    ;LO BYTE OF LOG FLAG/INT REG S/B=30
14952
14953  060412  012737  060422  000020    3$:     MOV     #4$,@#20        ;LOAD IOT VECTOR WITH 4$
14954  060420  000004                            IOT                     ;SECOND INTERRUPT-SHOULD LOAD LOG FLAG REG
14955  060422  022626                    4$:     CMP     (SP)+,(SP)+     ;CLEANUP STACK
14956  060424  012737  061204  000020            MOV     #$SCOPE,@#20    ;RESTORE IOT VECTOR
14957  060432  012737  060444  000004            MOV     #5$,@#4         ;SET UP CPU VECTOR
14958  060440  005737  160000                    TST     @#160000        ;FORCE TIMEOUT
14959  060444  022626                    5$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14960  060446  012737  061144  000004            MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
14961  060454  076600                            MED                     ;CHECK FLAG
14962  060456  000104                            RDLFGINT
14963  060460  120027  000020                    CMPB    R0,#20          ;IOT VECTOR LAST LOGGED?
14964  060464  001401                            BEQ     6$              ;BR IF YES
14965  060466  104036                            ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
14966                                                                    ;LOW BYTE S/B = 20
14967
14968  060470  012700  000767            6$:     MOV     #$TN-1,R0       ;SET UP FOR MISSED TEST CHECK AND
14969                                                                    ;FULL WORD TEST NUMBER FOR APT
14970
14971                                    .ENABLE AMA
14972
14973
14974                                    .SBTTL  END OF PASS ROUTINE
14975
14976                                    ;;**********************************************************************
14977                                    ;*INCREMENT THE PASS NUMBER ($PASS)
14978                                    ;*IF THERES A MONITOR GO TO IT
14979                                    ;*IF THERE ISN'T JUMP TO INIT
14980
14981  060474                            $EOP:
14982  060474  000004                            SCOPE
14983  060476  005037  001110                    CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
14984  060502  005237  001126                    INC     $PASS           ;;INCREMENT THE PASS NUMBER
14985  060506  042737  100000  001126            BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
14986  060514  005327                            DEC     (PC)+           ;;LOOP?
14987  060516  000001                    $EOPCT: .WORD   1
14988  060520  003027                            BGT     $DOAGN          ;;YES
14989  060522  012737                            MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
14990  060524  000001                    $ENDCT: .WORD   1
14991  060526  060516                            $EOPCT
14992  060530  104401  065027                    TYPE    ,EOP1           ;TYPE "END PASS #"
14993  060534  013746  001126                    MOV     $PASS,-(SP)     ;SAVE $PASS FOR TYPOUT
14994  060540  104402                            TYPOC                   ;TYPE PASS NUMBER IN OCTAL
14995  060542  104401  065045                    TYPE    ,EOP2           ;TYPE "ERROR COUNT ="
14996  060546  013746  001012                    MOV     $ERTTL,-(SP)    ;SAVE ERROR TOTAL FOR TYPOUT
14997  060552  104402                            TYPOC                   ;TYPE ERROR TOTAL
14998  060554  104401  001115                    TYPE    ,$CRLF
14999  060560  013700  000042            $GET42: MOV     @#42,R0         ;;GET MONITOR ADDRESS
```

```
 15000  060564  001405                        BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
 15001  060566  000005                        RESET                   ;;CLEAR THE WORLD
 15002  060570  004710                $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
 15003  060572  000240                        NOP                     ;;SAVE ROOM
 15004  060574  000240                        NOP                     ;;FOR
 15005  060576  000240                        NOP                     ;;ACT11
 15006  060600                        $DOAGN:
 15007  060600  000137                        JMP     @(PC)+          ;;RETURN
 15008  060602  003236                $RTNAD: .WORD   INIT
 15009  060604     377      377  000 $ENULL: .BYTE   -1,-1,0         ;;NULL CHARACTER STRING
 15010          060610                        .EVEN
```

```
15011                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15012                                           .SBTTL  /  /  /  /  /  /  UTILITIES  /  /  /  /  /  /
15013                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15014
15015                                   .SBTTL  POWER DOWN AND UP ROUTINES
15016
15017                                   ;;*****************************************************************
15018                                   ;POWER DOWN ROUTINE
15019  060610  012737  060762  000024  $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
15020  060616  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15021  060624  010046                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
15022  060626  010146                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
15023  060630  010246                  MOV     R2,-(SP)        ;;PUSH R2 ON STACK
15024  060632  010346                  MOV     R3,-(SP)        ;;PUSH R3 ON STACK
15025  060634  010446                  MOV     R4,-(SP)        ;;PUSH R4 ON STACK
15026  060636  010546                  MOV     R5,-(SP)        ;;PUSH R5 ON STACK
15027  060640  017746  120174          MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
15028  060644  010637  060766          MOV     SP,$SAVR6       ;;SAVE SP
15029  060650  012737  060662  000024  MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
15030  060656  000000                  HALT
15031  060660  000776                  BR      .-2             ;;HANG UP
15032
15033                                   ;;*****************************************************************
15034                                   ;POWER UP ROUTINE
15035  060662  012737  060762  000024  $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
15036  060670  013706  060766          MOV     $SAVR6,SP       ;;GET SP
15037  060674  005037  060766          CLR     $SAVR6          ;;WAIT LOOP FOR THE TTY
15038  060700  005237  060766  1$:     INC     $SAVR6          ;;WAIT FOR THE INC
15039  060704  001375                  BNE     1$              ;;OF  WORD
15040  060706  011600                  MOV     (SP),R0         ;GET OLD SWR VALUE
15041  060710  076600                  MED                     ;WRITE BACK ORIGINAL SWR VALUE
15042  060712  000226                  WCNSSW                  ;INTO HARDWARE SWITCH REGISTER
15043  060714  012677  120120          MOV     (SP)+,@SWR      ;;POP STACK INTO @SWR
15044  060720  012605                  MOV     (SP)+,R5        ;;POP STACK INTO R5
15045  060722  012604                  MOV     (SP)+,R4        ;;POP STACK INTO R4
15046  060724  012603                  MOV     (SP)+,R3        ;;POP STACK INTO R3
15047  060726  012602                  MOV     (SP)+,R2        ;;POP STACK INTO R2
15048  060730  012601                  MOV     (SP)+,R1        ;;POP STACK INTO R1
15049  060732  012600                  MOV     (SP)+,R0        ;;POP STACK INTO R0
15050  060734  012737  060610  000024  MOV     #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
15051  060742  012737  000340  000026  MOV     #340,@#PWRVEC+2 ;;PRIO:7
15052  060750  104401                  TYPE                    ;REPORT THE POWER FAILURE
15053  060752  060770          $PWRMG: .WORD   $POWER           ;;POWER FAIL MESSAGE POINTER
15054  060754  012716                  MOV     (PC)+,(SP)      ;;RESTART AT PWRUP
15055  060756  061000          $PWRAD: .WORD   PWRUP           ;;RESTART ADDRESS
15056  060760  000002                  RTI
15057  060762  000000          $ILLUP: HALT                    ;;THE POWER UP SEQUENCE WAS STARTED
15058  060764  000776                  BR      .-2             ;; BEFORE THE POWER DOWN WAS COMPLETE
15059  060766  000000          $SAVR6: 0                       ;;PUT THE SP HERE
15060  060770  005015  047520  042527  $POWER: .ASCIZ  <15><12>"POWER"
15061  060776  000122
15062                                   .EVEN
15063
15064  061000  012706  001000  PWRUP:  MOV     #STACK,SP       ;RESET SP
15065  061004  005037  177776          CLR     @#PSW           ;PRIORITY 0 -- CLEAR CODES
15066  061010  000137  001630          JMP     @#START         ;RESTART PROGRAM
```

```
15067
15068                                   ; ***************************************************************************
15069                                           .SBTTL "T" BIT SERVICE ROUTINE
15070                                   ; ***************************************************************************
15071
15072  061014  062716  000002          TBSER:  ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
15073  061020  042766  000020  000002          BIC     #20,2(SP)       ;TURN OFF THE "T" BIT
15074  061026  000006                          RTT                     ;RETURN TO THE CALLING TEST
15075
15076                                   .SBTTL  MICROBREAK TRAP SERVICE ROUTINE
15077                                   ;;*************************************************************
15078                                   ;*      THIS ROUTINE MERELY SETS A FLAG
15079                                   ;*      WHEN THE ROUTINE HAS BEEN ENTERED
15080                                   ;*
15081  061030  005237  061036          BKROUT: INC     BKFLAG          ;SET MICROBREAK FLAG TO
15082                                                                   ;INDICATE TRAP TO 4 OCCURRED
15083  061034  000002                          RTI                     ;RETURN FROM TRAP
15084  061036  000000                  BKFLAG: .WORD   0               ;MICROBREAK TRAP FLAG
15085
15086                                   ; ***************************************************************************
15087                                           .SBTTL  RSVD INSTRUCTION TRAP SERVICE ROUTINE
15088                                   ; ***************************************************************************
15089
15090                                   ;THIS ROUTINE SERVICES UNEXPECTED RESERVED INSTRUCTION TRAP ERRORS
15091                                   ;IT RESULTS IN PRINTING THE ERROR MESSAGE: "TRAPPED TO 10 PC=XXXXXX"
15092                                   ;WHERE XXXXXX IS THE ADDRESS CONTAINING THE INSTRUCTION WORD THAT
15093                                   ;SPRUNG THE TRAP.  AFTER PRINTING THE ERROR MESSAGE AN ATTEMPT IS
15094                                   ;MADE TO RESTART THE PROGRAM AT THE BEGINNING.
15095
15096                                   ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A
15097                                   ;PREVIOUS RSVD INSTRUCTION TRAP OR AN UNEXPECTED BUS ERROR THE PROGRAM
15098                                   ;WILL HALT.  AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE
15099                                   ;TO THE TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15100
15101                                           ;(SP)   PC+2    OF 2ND TRAP
15102                                           ;(SP)+2 PSW       "
15103                                           ;(SP)+4 PC+2    OF 1ST TRAP
15104                                           ;(SP)+6 PSW       "
15105
15106                                           ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE FOLLOWING
15107                                           ;INFORMATION:
15108
15109                                                   ;(CATERR)=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15110                                                                   ;TRAP (PC AT TIME OF ERROR HALT INDICATES
15111                                                                   ;WHICH OCCURRED FIRST)
15112                                                   ;(CATERR)=2     TWO SUCCESSIVE BUS ERROR TRAPS
15113                                                   ;(CATERR)=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15114
15115                                   ;THE CONTENTS OF R0 AT THE TIME OF THE
15116                                   ;HALT PROVIDES FURTHER INFORMATION AS TO THE LAST TEST BEING EXECUTED
15117                                   ;WHEN THE TRAPS OCCURRED.
15118
15119                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION
15120                                   ;TESTS TO VERIFY THE RSVD INSTR TRAP MECHANISM PRIOR TO ACTIVATING THE SERVICE
15121                                   ;ROUTINE
15122
```

```
15123  061040  005137  063172          RSVTST: COM     RSVFLG          ;SET RSVD INSTR TRAP TEST FLAG
15124  061044  000002                          RTI                     ;RETURN TO BASIC TEST
15125
15126  061046  005737  063176          RSERR:  TST     @#CATERR        ;ANY PENDING CATASTROPHIC ERRORS
15127  061052  001025                          BNE     INCRSV          ;BE IF YES
15128  061054  105237  063177                  INCB    @#1+CATERR      ;SET RSVD INSTR FLAG
15129  061060  032777  010000  117752          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT ?
15130  061066  001015                          BNE     RESTAR          ;BR IF YES
15131  061070  104401                          TYPE                    ;GO TYPE "TRAPPED TO 10 PC="
15132  061072  065167                          RSMSG
15133  061074  011646          RSBERT: MOV     (SP),-(SP)      ;GET ERROR PC ON STACK FOR PRINTING
15134  061076  104402                          TYPOC                   ;TYPE THE ERROR PC
15135  061100  104401                          TYPE                    ;OUTPUT CR / LF
15136  061102  001115                          $CRLF
15137  061104  005237  001012                  INC     @#SERTTL        ;COUNT THE ERROR
15138  061110  032777  100000  117722          BIT     #BIT15,@SWR     ;HALT ON ERROR?
15139  061116  001401                          BEQ     RESTAR          ;BR IF NOT
15140  061120  000000                          HALT                    ;HALT ON ERROR--PRESS CONTINUE TO RESTART
15141  061122  000137  003236          RESTAR: JMP     @#INIT          ;GO ATTEMPT RESTART
15142  061126  105237  063177          INCRSV: INCB    @#1+CATERR      ;INCREMENT RSVD INSTR FLAG
15143  061132  000000                          HALT                    ;CATASTROPHIC ERROR HALT
15144  061134  000772                          BR      RESTAR          ;DEPRESSING CONTINUE WILL CAUSE
15145                                                                  ;ATTEMPT TO RESTART.
15146
15147                          ; ***************************************************************************
15148                                  .SBTTL  BUS ERROR TRAP SERVICE ROUTINE
15149                          ; ***************************************************************************
15150
15151                          ;THIS ROUTINE SERVICES UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD ADDRESS
15152                          ;ERRORS, STACK OVERFLOW, AND ILLEGAL INSTRUCTIONS). IT RESULTS IN PRINTING THE
15153                          ;ERROR MESSAGE: "TRAPPED TO 4 PC =XXXXXX" WHERE XXXXXX IS THE
15154                          ;CONTENTS OF THE PC WHEN THE TRAP WAS SPRUNG.  AFTER PRINTING THE
15155                          ;ERROR MESSAGE AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
15156                          ;THE BEGINNING.
15157
15158                          ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A PREVIOUS
15159                          ;RSVD INSTR TRAP OR A PREVIOUS BUS ERROR, THE PROGRAM WILL HALT.
15160                          ;AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE TO THE
15161                          ;TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15162
15163                                  ;[SP]   PC+2    OF 2ND TRAP
15164                                  ;[SP]+2 PSW       "
15165                                  ;[SP]+4 PC+2    OF 1ST TRAP
15166                                  ;[SP]+6 PSW       "
15167
15168                                  ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE LOLLOWING
15169                                  ;INFORMATION:
15170
15171                                          ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15172                                                          ;TRAP (PC AT TIME OF ERROR HALT
15173                                                          ;INDICATES WHICH OCCURRED FIRST)
15174                                          ;[CATERR]=2     TWO SUCCESSIVE BUS ERRORS
15175                                          ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15176
15177                          ;THE CONTENTS OF R0 AT THE TIME OF
15178                          ;THE HALT PROVIDED FURTHER INFORMATION AS TO THE TEST IN PROGRESS
```

```
15179                                   ;WHEN THE TRAPS OCCURRED.
15180
15181                                   ;THE CONTENTS OF THE SP CAN BE USED TO INDICATE IF STACK OVERFLOW CAUSED
15182                                   ;THE BUSS ERROR TRAP(S) AS SHOWN BELOW:
15183
15184                                           ;400>(SP)>336   YELLOW ZONE
15185                                           ;(SP)=0         RED ZONE
15186
15187                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION TESTS TO
15188                                   ;VERIFY THAT THE BUS ERROR TRAP MECHANISM WORKS PRIOR TO ACTIVATING
15189                                   ;THE SERVICE ROUTINE
15190
15191  061136  005137  063174         BETST:  COM     BERFLG          ;SET BUS ERROR TRAP TEST FLAG
15192  061142  000002                         RTI                     ;RETURN TO BASIC TEST
15193
15194  061144  005737  063176         BERR:   TST     @#CATERR        ;ANY CATASTROPHIC ERRORS PENDING?
15195  061150  001011                         BNE     2$              ;BR IF YES
15196  061152  105237  063176                 INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15197  061156  032777  010000  117654         BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT
15198  061164  001356                         BNE     RESTAR          ;BR IF YES
15199  061166  104401                         TYPE                    ;PRINT "TRAP TO 4" MESSAGE
15200  061170  065142                         BEMSG
15201  061172  000740                         BR      RSBERT          ;TYPE REST OF BUS ERROR MESSAGE
15202
15203  061174  105237  063176         2$:     INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
15204  061200  000000                         HALT                    ;CATASTROPHIC ERROR HALT-SCHOOLS OUT
15205  061202  000747                         BR      RESTAR          ;DEPRESS CONTINUE TO ATTEMPT RESTART
15206
15207
15208
15209                                   .SBTTL  SCOPE HANDLER ROUTINE
15210
15211                                   ;;**************************************************************
15212                                   ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
15213                                   ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
15214                                   ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
15215                                   ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
15216                                   ;*SW14=1        LOOP ON TEST
15217                                   ;*SW11=1        INHIBIT ITERATIONS
15218                                   ;*SW09=1        LOOP ON ERROR
15219                                   ;*CALL
15220                                   ;*      SCOPE           ;;SCOPE=IOT
15221
15222  061204                         $SCOPE:
15223  061204  020037  001124                 CMP     R0,@#$TESTN     ;ANY MISSED TESTS ?
15224  061210  001406                         BEQ     10$             ;BR IF NOT
15225  061212  012737  061222  001112         MOV     #12$,@#$ESCAPE  ;NO ERROR LOOPING
15226  061220  104011                         ERROR   11              ;MISSED TESTS ERROR CALL
15227  061222  005037  001112         12$:    CLR     @#$ESCAPE       ;NORMAL ERROR LOOPING
15228  061226  110037  001002         10$:    MOVB    R0,@#$TSTNM     ;INSURE TSTNUM IS CORRECT
15229  061232  032777  002000  117600         BIT     #SW10,@SWR      ;LOOP ON SELECTED TEST?
15230  061240  001411                         BEQ     11$             ;BR IF NO
15231  061242  017737  117572  063166         MOV     @SWR,@#SELTST   ;GET CONTENTS OF SWITCHES
15232  061250  042737  177000  063166         BIC     #177000,@#SELTST        ;MASK OUT SWR<15:9>
15233  061256  020037  063166                 CMP     R0,@#SELTST     ;IS THIS THE SELECTED TEST?
15234  061262  001510                         BEQ     $OVER           ;BR IF YES
```

```
15235  061264                                  11$:
15236  061264  032777  040000  117546  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
15237  061272  001104                          BNE     $OVER           ;;YES IF SW14=1
15238                                          ;#####START OF CODE FOR THE XOR TESTER#####
15239  061274  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE "XOR" TESTER CHANGE
15240                                                                  ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
15241  061276  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
15242  061302  012737  061322  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
15243  061310  005737  177060                  TST     @#177060        ;;TIME OUT ON XOR?
15244  061314  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15245  061320  000453                          BR      $SVLAD          ;;GO TO THE NEXT TEST
15246  061322  022626                  5$:     CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
15247  061324  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15248  061330  000413                          BR      7$              ;;LOOP ON THE PRESENT TEST
15249  061332                                  6$:;#####END OF CODE FOR THE XOR TESTER#####
15250  061332  105737  001003          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
15251  061336  001421                          BEQ     3$              ;;BR IF NO
15252  061340  123737  001015  001003          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
15253  061346  101015                          BHI     3$              ;;BR IF NO
15254  061350  032777  001000  117462          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
15255  061356  001404                          BEQ     4$              ;;BR IF NO
15256  061360  013737  001010  001006  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
15257  061366  000446                          BR      $OVER
15258  061370  105037  001003          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
15259  061374  005037  001110                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
15260  061400  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
15261  061402  032777  004000  117430  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
15262  061410  001011                          BNE     1$              ;;BR IF YES
15263  061412  005737  001126                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
15264  061416  001406                          BEQ     1$              ;;      INHIBIT ITERATIONS
15265  061420  005237  001004                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
15266  061424  023737  001110  001004          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
15267  061432  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
15268  061434  012737  000001  001004  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
15269  061442  013737  061534  001110          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
15270  061450  105237  001002          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
15271  061454  113737  001002  001124          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
15272  061462  011637  001006                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
15273  061466  011637  001010                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
15274  061472  005037  001112                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
15275  061476  112737  000001  001015          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
15276  061504  013777  001002  117330  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
15277  061512  013716  001006                  MOV     $LPADR,(SP)     ;FUDGE RETURN ADDRESS
15278  061516  120037  001002                  CMPB    R0,@#$TSTNM     ;WAS $TSTNM INCREMENTED?
15279  061522  001401                          BEQ     10$             ;BR IF NOT
15280  061524  005200                          INC     R0              ;INCREMENT TEST NUMBER
15281  061526  010037  001124          10$:    MOV     R0,@#$TESTN     ;FIX $TESTN TO BE WORD COUNT, NOT BYTE
15282  061532  000002                          RTI
15283  061534  000200                  $MXCNT: 200                     ;;MAX. NUMBER OF ITERATIONS
15284
15285  061536  005137  063170          SCOPEA: COM     @#SCOFLG        ;THESE TWO INSTRUCTIONS ARE
15286  061542  000002                          RTI                     ;USED IN THE BASIC TESTS TO
15287                                                                  ;VERIFY THE IOT LINKAGE
15288
15289
15290
```

```
 15291                                  .SBTTL  ERROR HANDLER ROUTINE
 15292
 15293                                  ;;**********************************************************************
 15294                                  ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
 15295                                  ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
 15296                                  ;*AND GO TO $ERRTYP ON ERROR
 15297                                  ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
 15298                                  ;*SW15=1          HALT ON ERROR
 15299                                  ;*SW13=1          INHIBIT ERROR TYPEOUTS
 15300                                  ;*SW09=1          LOOP ON ERROR
 15301                                  ;*CALL
 15302                                  ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER
 15303
 15304  061544                          $ERROR:
 15305  061544  010546                         MOV     R5,-(SP)        ;SAVE R5 ON STACK
 15306  061546  012705  001060                 MOV     #$REGAD,R5      ;GET POINTER
 15307  061552  016625  000004                 MOV     4(SP),(R5)+     ;SAVE ERROR PSW IN $REGAD FOR TYPOUT
 15308  061556  010025                         MOV     R0,(R5)+        ;SAVE R0 FOR TYPEOUTS
 15309  061560  010125                         MOV     R1,(R5)+        ;SAVE R1 IN $REG1
 15310  061562  010225                         MOV     R2,(R5)+        ;SAVE R2 IN $REG2, ETC.
 15311  061564  010325                         MOV     R3,(R5)+
 15312  061566  010425                         MOV     R4,(R5)+
 15313  061570  022715  177777                 CMP     #-1,(R5)        ;IS SP  ALREADY STORED IN $REG5?
 15314  061574  001001                         BNE     10$             ;BR IF YES
 15315  061576  010615                         MOV     SP,(R5)         ;PUT SP IN $REG5 FOR TYPOUT
 15316  061600  012605                 10$:    MOV     (SP)+,R5        ;RESTORE R5
 15317  061602  105237  001003         7$:     INCB    @#ERFLG         ;;SET THE ERROR FLAG
 15318  061606  001775                         BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
 15319  061610  013777  001002  117224         MOV     $TSTNM,@#DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
 15320  061616  005237  001012                 INC     $ERTTL          ;;INC THE ERROR COUNT
 15321  061622  011637  001016                 MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
 15322  061626  162737  000002  001016         SUB     #2,$ERRPC
 15323  061634  117737  117156  001014         MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
 15324  061642  032777  020000  117170         BIT     #BIT13,@#SWR    ;;SKIP TYPEOUT IF SET
 15325  061650  001004                         BNE     20$             ;;SKIP TYPEOUTS
 15326  061652  004737  061772                 JSR     PC,$ERRTYP      ;;GO TO USER ERROR ROUTINE
 15327  061656  104401  001115                 TYPE    ,$CRLF
 15328  061662                          20$:
 15329  061662  122737  000001  001140         CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
 15330  061670  001007                         BNE     2$              ;;NO,SKIP APT ERROR REPORT
 15331  061672  113737  001014  061704         MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
 15332  061700  004737  062662                 JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
 15333  061704     000                  21$:    .BYTE   0
 15334  061705     000                          .BYTE   0
 15335  061706  000777                 22$:    BR      22$             ;;APT ERROR LOOP
 15336  061710  005777  117124         2$:     TST     @#SWR           ;;HALT ON ERROR
 15337  061714  100001                         BPL     3$              ;;SKIP IF CONTINUE
 15338  061716  000000                         HALT                    ;;HALT ON ERROR!
 15339  061720  032777  001000  117112 3$:     BIT     #BIT09,@#SWR    ;;LOOP ON ERROR SWITCH SET?
 15340  061726  001402                         BEQ     4$              ;;BR IF NO
 15341  061730  013716  001010                 MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
 15342  061734  005737  001112         4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
 15343  061740  001402                         BEQ     5$              ;;BR IF NONE
 15344  061742  013716  001112                 MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
 15345  061746                          5$:
 15346  061746  012737  177777  001074         MOV     #-1,@#$REGS     ;FLAG CURRENT STACK POINTER TO BE TYPED
```

```
15347  061754  042766  000020  000002          BIC     #20,2(SP)       ;CLEAR T BIT IN CASE ERROR OCCURED
15348                                                                   ;IN T BIT TESTS
15349  061762  000002                          RTI
15350
15351  061764  005137  063164          ERRA:   COM     @#ERRFLG        ;THESE TWO INSTRUCTIONS ARE USED
15352  061770  000002                          RTI                     ;IN THE BASIC TESTS TO VERIFY THE EMT
15353
15354                                  .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
15355
15356                                  ;;**********************************************************************
15357                                  ;*THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
15358                                  ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE "ERROR TABLE" ($ERRTB),
15359                                  ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
15360
15361  061772                          $ERRTYP:
15362  061772  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
15363  061776  010046                          MOV     R0,-(SP)        ;;SAVE R0
15364  062000  005000                          CLR     R0              ;;PICKUP THE ITEM INDEX
15365  062002  153700  001014                  BISB    @#$ITEMB,R0
15366  062006  001004                          BNE     1$              ;;IF ITEM NUMBER IS ZERO, JUST
15367                                                                   ;;TYPE THE PC OF THE ERROR
15368  062010  013746  001016                  MOV     $ERRPC,-(SP)    ;;SAVE $ERRPC FOR TYPEOUT
15369                                                                   ;;ERROR ADDRESS
15370  062014  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
15371  062016  000426                          BR      6$              ;;GET OUT
15372  062020  005300                  1$:     DEC     R0              ;;ADJUST THE INDEX SO THAT IT WILL
15373  062022  006300                          ASL     R0              ;;      WORK FOR THE ERROR TABLE
15374  062024  006300                          ASL     R0
15375  062026  006300                          ASL     R0
15376  062030  062700  001150                  ADD     #$ERRTB,R0      ;;FORM TABLE POINTER
15377  062034  012037  062044                  MOV     (R0)+,2$        ;;PICKUP "ERROR MESSAGE" POINTER
15378  062040  001404                          BEQ     3$              ;;SKIP TYPEOUT IF NO POINTER
15379  062042  104401                          TYPE                    ;;TYPE THE "ERROR MESSAGE"
15380  062044  000000                  2$:     .WORD   0               ;;"ERROR MESSAGE" POINTER GOES HERE
15381  062046  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
15382  062052  012037  062062          3$:     MOV     @(R0)+,4$       ;;PICKUP "DATA HEADER" POINTER
15383  062056  001404                          BEQ     5$              ;;SKIP TYPEOUT IF 0
15384  062060  104401                          TYPE                    ;;TYPE THE "DATA HEADER"
15385  062062  000000                  4$:     .WORD   0               ;;"DATA HEADER" POINTER GOES HERE
15386  062064  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
15387  062070  011000                  5$:     MOV     @(R0),R0        ;;PICKUP "DATA TABLE" POINTER
15388  062072  001004                          BNE     7$              ;;GO TYPE THE DATA
15389  062074  012600                  6$:     MOV     (SP)+,R0        ;;RESTORE R0
15390  062076  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
15391  062102  000207                          RTS     PC              ;;RETURN
15392  062104                          7$:
15393  062104  013046                          MOV     @(R0)+,-(SP)    ;;SAVE @(R0)+ FOR TYPEOUT
15394  062106  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
15395  062110  005710                          TST     (R0)            ;;IS THERE ANOTHER NUMBER?
15396  062112  001770                          BEQ     6$              ;;BR IF NO
15397  062114  104401  062122                  TYPE    ,8$             ;;TYPE TWO(2) SPACES
15398  062120  000771                          BR      7$              ;;LOOP
15399  062122  020040     000          8$:     .ASCIZ  /  /            ;;TWO(2) SPACES
15400           062126                          .EVEN
15401
15402                                  ; **********************************************************************
```

```
15403                                           .SBTTL PRINT ROUTINES
15404                                   ; ******************************************************************
15405
15406  062126  005137  063162          PRINA:  COM     @#PRIFLG        ;THESE TWO INSTRUCTIONS ARE
15407  062132  000002                          RTI                     ;USED BY THE BASIC TESTS TO VERIFY
15408                                                                  ;THE TRAP INSTRUCTION
15409
15410                                   .SBTTL  TYPE ROUTINE
15411
15412                                   ;;******************************************************************
15413                                   ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
15414                                   ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
15415                                   ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
15416                                   ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
15417                                   ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
15418                                   ;*
15419                                   ;*CALL:
15420                                   ;*1) USING A TRAP INSTRUCTION
15421                                   ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
15422                                   ;*OR
15423                                   ;*      TYPE
15424                                   ;*      MESADR
15425                                   ;*
15426
15427  062134  105737  001057          $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
15428  062140  100002                          BPL     1$              ;;BR IF YES
15429  062142  000000                          HALT                    ;;HALT HERE IF NO TERMINAL
15430  062144  000430                          BR      3$              ;;LEAVE
15431  062146  010046                  1$:     MOV     R0,-(SP)        ;;SAVE R0
15432  062150  017600  000002                  MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
15433  062154  122737  000001  001140          CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
15434  062162  001011                          BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
15435  062164  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
15436  062172  001405                          BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
15437  062174  010037  062204                  MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
15438  062200  004737  062652                  JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
15439  062204  000000                  61$:    .WORD   0               ;;MESSAGE ADDRESS
15440  062206  132737  000040  001141  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
15441  062214  001003                          BNE     60$             ;;YES,SKIP TYPE OUT
15442  062216  112046                  2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
15443  062220  001005                          BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
15444  062222  005726                          TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
15445  062224  012600                  60$:    MOV     (SP)+,R0        ;;RESTORE R0
15446  062226  062716  000002          3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
15447  062232  000002                          RTI                     ;;RETURN
15448  062234  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
15449  062240  001430                          BEQ     8$
15450  062242  122716  000200                  CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
15451  062246  001006                          BNE     5$
15452  062250  005726                          TST     (SP)+           ;;POP  <CR><LF> EQUIV
15453  062252  104401                          TYPE                    ;;TYPE A CR AND LF
15454  062254  001115                          $CRLF
15455  062256  105037  062412                  CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
15456  062262  000755                          BR      2$              ;;GET NEXT CHARACTER
15457  062264  004737  062346          5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
15458  062270  123726  001056          6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
```

```
15459  062274  001350                        BNE     2$              ;;IF NO GO GET NEXT CHAR.
15460  062276  013746  001054                MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
15461                                                                 ;;AND THE NULL CHAR.
15462  062302  105366  000001        7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
15463  062306  002770                        BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
15464  062310  004737  062346                JSR     PC,$TYPEC       ;;GO TYPE A NULL
15465  062314  105337  062412                DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
15466  062320  000770                        BR      7$              ;;LOOP
15467
15468                                ;HORIZONTAL TAB PROCESSOR
15469
15470  062322  112716  000040        8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
15471  062326  004737  062346        9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
15472  062332  132737  000007  062412        BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
15473  062340  001372                        BNE     9$              ;;TAB STOP
15474  062342  005726                        TST     (SP)+           ;;POP SPACE OFF STACK
15475  062344  000724                        BR      2$              ;;GET NEXT CHARACTER
15476  062346  105777  116476        $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
15477  062352  100375                        BPL     $TYPEC
15478  062354  116677  000002  116470        MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
15479  062362  122766  000015  000002        CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
15480  062370  001003                        BNE     1$              ;;BRANCH IF NO
15481  062372  105037  062412                CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
15482  062376  000406                        BR      $TYPEX          ;;EXIT
15483  062400  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
15484  062406  001402                        BEQ     $TYPEX          ;;BRANCH IF YES
15485  062410  105227                        INCB    (PC)+           ;;COUNT THE CHARACTER
15486  062412  000000        $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
15487  062414  000207        $TYPEX: RTS     PC
15488
15489
15490                                .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
15491
15492                                ;;**************************************************************
15493                                ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
15494                                ;*OCTAL (ASCII) NUMBER AND TYPE IT.
15495                                ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
15496                                ;*CALL:
15497                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
15498                                ;*      TYPOS                   ;;CALL FOR TYPEOUT
15499                                ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
15500                                ;*      .BYTE   M               ;;M=1 OR 0
15501                                ;*                                      ;;1=TYPE LEADING ZEROS
15502                                ;*                                      ;;0=SUPPRESS LEADING ZEROS
15503                                ;*
15504                                ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
15505                                ;*$TYPOS OR $TYPOC
15506                                ;*CALL:
15507                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
15508                                ;*      TYPON                   ;;CALL FOR TYPEOUT
15509                                ;*
15510                                ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
15511                                ;*CALL:
15512                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
15513                                ;*      TYPOC                   ;;CALL FOR TYPEOUT
15514
```

```
15515  062416  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
15516  062422  116637  000001  062641          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
15517  062430  112637  062643                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
15518  062434  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
15519  062440  000406                          BR      $TYPON
15520  062442  112737  000001  062641  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
15521  062450  112737  000006  062643          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
15522  062456  112737  000005  062640  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
15523  062464  010346                          MOV     R3,-(SP)        ;;SAVE R3
15524  062466  010446                          MOV     R4,-(SP)        ;;SAVE R4
15525  062470  010546                          MOV     R5,-(SP)        ;;SAVE R5
15526  062472  113704  062643                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
15527  062476  005404                          NEG     R4
15528  062500  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
15529  062504  110437  062642                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
15530  062510  113704  062641                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
15531  062514  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
15532  062520  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
15533  062522  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
15534  062524  000404                          BR      3$              ;;GO DO MSB
15535  062526  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
15536  062530  006105                          ROL     R5
15537  062532  006105                          ROL     R5
15538  062534  010503                          MOV     R5,R3
15539  062536  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
15540  062540  105337  062642                  DECB    $OMODE          ;;TYPE THIS DIGIT?
15541  062544  100016                          BPL     7$              ;;BR IF NO
15542  062546  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
15543  062552  001002                          BNE     4$              ;;TEST FOR 0
15544  062554  005704                          TST     R4              ;;SUPPRESS THIS 0?
15545  062556  001403                          BEQ     5$              ;;BR IF YES
15546  062560  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
15547  062562  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
15548  062566  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
15549  062572  110337  062636                  MOVB    R3,8$           ;;SAVE FOR TYPING
15550  062576  104401  062636                  TYPE    ,8$             ;;GO TYPE THIS DIGIT
15551  062602  105337  062640          7$:     DECB    $OCNT           ;;COUNT BY 1
15552  062606  003347                          BGT     2$              ;;BR IF MORE TO DO
15553  062610  002402                          BLT     6$              ;;BR IF DONE
15554  062612  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
15555  062614  000744                          BR      2$              ;;GO DO THE LAST DIGIT
15556  062616  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
15557  062620  012604                          MOV     (SP)+,R4        ;;RESTORE R4
15558  062622  012603                          MOV     (SP)+,R3        ;;RESTORE R3
15559  062624  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
15560  062632  012616                          MOV     (SP)+,(SP)
15561  062634  000002                          RTI                     ;;RETURN
15562  062636     000          8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
15563  062637     003                  .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
15564  062640     000          $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
15565  062641     000          $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
15566  062642  000000          $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
15567
15568                          .SBTTL  APT COMMUNICATIONS ROUTINE
15569
15570                          ;;**************************************************************
```

```
15571  062644  112737  000001  063110  $ATY1:  MOVB    #1,$FFLG        ;;TO REPORT FATAL ERROR
15572  062652  112737  000001  063106  $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
15573  062660  000403                  BR      $ATYC
15574  062662  112737  000001  063110  $ATY4:  MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
15575  062670                          $ATYC:
15576  062670  010046                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
15577  062672  010146                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
15578  062674  105737  063106          TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
15579  062700  001450                  BEQ     5$              ;;IF NOT:  BR
15580  062702  122737  000001  001140  CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
15581  062710  001031                  BNE     3$              ;;IF NOT:  BR
15582  062712  132737  000100  001141  BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
15583  062720  001425                  BEQ     3$              ;;IF NOT:  BR
15584  062722  017600  000004          MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
15585  062726  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
15586  062734  005737  001120          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
15587  062740  001375                  BNE     1$              ;;IF NOT:  WAIT
15588  062742  010037  001134          MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
15589  062746  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
15590  062750  001376                  BNE     2$
15591  062752  163700  001134          SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
15592  062756  006200                  ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
15593  062760  010037  001136          MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
15594  062764  012737  000004  001120  MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
15595  062772  000413                  BR      5$
15596  062774  017637  000004  063020  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
15597  063002  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
15598  063010  013746  177776          MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
15599  063014  004737  062134          JSR     PC,$TYPE        ;;CALL TYPE MACRO
15600  063020  000000                  4$:     .WORD   0
15601  063022                          5$:
15602  063022  105737  063110          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
15603  063026  001416                  BEQ     12$             ;;IF NOT:  BR
15604  063030  005737  001140          TST     $ENV            ;;RUNNING UNDER APT?
15605  063034  001413                  BEQ     12$             ;;IF NOT:  BR
15606  063036  005737  001120          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
15607  063042  001375                  BNE     11$             ;;IF NOT:  WAIT
15608  063044  017637  000004  001122  MOV     @4(SP),$FATAL   ;;GET ERROR #
15609  063052  062766  000002  000004  ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
15610  063060  005237  001120          INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
15611  063064  105037  063110          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
15612  063070  105037  063107          CLRB    $LFLG           ;;CLEAR LOG FLAG
15613  063074  105037  063106          CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
15614  063100  012601                  MOV     (SP)+,R1        ;;POP STACK INTO R1
15615  063102  012600                  MOV     (SP)+,R0        ;;POP STACK INTO R0
15616  063104  000207                  RTS     PC              ;;RETURN
15617  063106     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
15618  063107     000          $LFLG:  .BYTE   0               ;;LOG FLAG
15619  063110     000          $FFLG:  .BYTE   0               ;;FATAL FLAG
15620          063112                  .EVEN
15621          000200          APTSIZE=200
15622          000001          APTENV=001
15623          000100          APTSPOOL=100
15624          000040          APTCSUP=040
15625
15626                          .SBTTL  TRAP DECODER
```

```
15627
15628                                   ;;**********************************************************************
15629                                   ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
15630                                   ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
15631                                   ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
15632                                   ;*GO TO THAT ROUTINE.
15633
15634  063112  010046                   $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
15635  063114  016600  000002                   MOV     2(SP),R0        ;;GET TRAP ADDRESS
15636  063120  005740                           TST     -(R0)           ;;BACKUP BY 2
15637  063122  111000                           MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
15638  063124  006300                           ASL     R0              ;;POSITION FOR INDEXING
15639  063126  016000  063146                   MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
15640  063132  000200                           RTS     R0              ;;GO TO ROUTINE
15641
15642
15643                                   ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
15644
15645  063134  011646                   $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
15646  063136  016666  000004  000002           MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
15647  063144  000002                           RTI                     ;;RESTORE THE PSW
15648
15649                                   .SBTTL  TRAP TABLE
15650
15651                                   ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
15652                                   ;*BY THE "TRAP" INSTRUCTION.
15653
15654                                   ;       ROUTINE
15655                                   ;       -------
15656  063146  063134                   $TRPAD: .WORD   $TRAP2
15657  063150  062134                           $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
15658  063152  062442                           $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
15659  063154  062416                           $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
15660  063156  062456                           $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
15661
15662
15663
15664                                   ;FLAGS, CONSTANTS, AND VARIABLES
15665
15666  063160  000000                   BPTLOC: 0                       ;STORES 16 USER DEFINED MAINTENANCE
15667                                                                   ;BREAKPOINTS
15668  063162  000000                   PRIFLG: 0                       ;FLAG USED BY BASIC TESTS FOR TRAP TEST
15669  063164  000000                   ERRFLG: 0                       ;FLAG USED BY BASIC TESTS FOR EMT TEST
15670  063166  000000                   SELTST: 0                       ;STORES SR<8:0> FOR LOOP ON SELECTED TEST
15671  063170  000000                   SCOFLG: 0                       ;USED BY BASIC TESTS FOR IOT TEST
15672  063172  000000                   RSVFLG: 0                       ;FLAG USED BY BASIC TEST OF RSVD INSTR TRAP
15673  063174  000000                   BERFLG: 0                       ;FLAG USED BY BASIC TEST OF BUS ERROR TRAPS
15674  063176  000000                   CATERR: 0                       ;FLAGS USED BY BUS ERROR AND RSVD INSTR TRAP
15675                                                                   ;SERVICE ROUTINES
15676  063200  000000                   ONCE:   0                       ;FLAGS PROGRAM TITLE HAS BEEN PRINTED
15677                                   ;COMMON DATA STRUCTURES AND MISCELLANEOUS TABLES
15678
15679  063202  177400                   OBUF:   177400                  ;DL11 OUTPUT TEST BUFFER
15680  063204  177400                           177400
15681  063206  177400                           177400
15682  063210  177400                           177400
```

```
15683
15684  063212  000004                  IBUF:   .BLKW   4               ;DL11 INPUT TEST BUFFER
15685
15686  063222  063246                  ATA:    DWTA
15687  063224  063756                          DWTB
15688  063226  064554                          D8TA
15689  063230  064560                          D8TB
15690  063232  063236                          MBUF0
15691  063234  063242                          MBUF1
15692
15693  063236  000000                  MBUF0:  0
15694  063240  000000                          0
15695  063242  000000                  MBUF1:  0
15696  063244  000000                          0
15697  063246  000000                  DWTA:   0
15698  063250  177777                          -1
15699  063252  177400                          177400
15700  063254  000377                          377
15701  063256  125252                          125252
15702  063260  052525                  ALUADD: 052525                  ;ALSO SERVES AS NULL ENTRY FOR ALUADD
15703
15704                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ADD TEST IN THE
15705                                  ;COMBINED INSTRUCTION TESTS
15706
15707  063262  000000                          000000                  ;SRC OP1
15708  063264  000000                          000000                  ;DST OP1
15709  063266  000000                          000000                  ;ANS1
15710  063270  177777                          177777                  ;SRC OP2
15711  063272  177777                          177777                  ;DST OP2
15712  063274  177776                          177776                  ;ANS2
15713  063276  125252                          125252                  ;SRC OP3
15714  063300  052525                          052525                  ;DST OP3
15715  063302  177777                          177777                  ;ANS3
15716  063304  052525                          052525                  ;SRC OP4
15717  063306  125252                          125252                  ;DST OP4
15718  063310  177777                          177777                  ;ANS4
15719  063312  125252                          125252                  ;SRC OP5
15720  063314  125252                          125252                  ;DST OP5
15721  063316  052524                          052524                  ;ANS5
15722  063320  052525                          052525                  ;SRC OP6
15723  063322  052525                          052525                  ;DST OP6
15724  063324  125252                          125252                  ;ANS6
15725  063326  052525                          052525                  ;SRC OP7
15726  063330  125253                          125253                  ;DST OP7
15727  063332  000000                          000000                  ;ANS7
15728  063334  125253                          125253                  ;SRC OP8
15729  063336  052525                          052525                  ;DST OP8
15730  063340  000000                  ANDTAB: 000000                  ;ANS8 -- ALSO NULL ENTRY FOR ANDTAB
15731
15732                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "AND" TESTS IN THE
15733                                  ;COMBINED INSTRUCTION EXERCISER TESTS
15734
15735  063342  000000                          000000                  ;SRC OP1
15736  063344  000000                          000000                  ;DEST OP1
15737  063346  000000                          000000                  ;ANS1
15738  063350  177777                          177777                  ;SRC OP2
```

```
15739  063352  177777                    177777                  ;DST OP2
15740  063354  000000                    000000                  ;ANS2
15741  063356  000000                    000000                  ;SRC OP3
15742  063360  177777                    177777                  ;DST OP3
15743  063362  177777                    177777                  ;ANS3
15744  063364  177777                    177777                  ;SRC OP4
15745  063366  000000                    000000                  ;DST OP4
15746  063370  000000                    000000                  ;ANS4
15747  063372  125252                    125252                  ;SRC OP5
15748  063374  125252                    125252                  ;DST OP5
15749  063376  000000                    000000                  ;ANS5
15750  063400  052525                    052525                  ;SRC OP6
15751  063402  052525                    052525                  ;DST OP6
15752  063404  000000                    000000                  ;ANS6
15753  063406  125252                    125252                  ;SRC OP7
15754  063410  052525                    052525                  ;DST OP7
15755  063412  052525                    052525                  ;ANS7
15756  063414  052525                    052525                  ;SRC OP8
15757  063416  125252                    125252                  ;DST OP8
15758  063420  125252          ORTAB:    125252                  ;ANS8 -- ALSO NULL ENTRY FOR ORTAB
15759
15760                          ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "OR" TEST IN THE
15761                          ;COMBINED INSTRUCTION EXERCISER TEST
15762
15763  063422  000000                    000000                  ;SRC OP1
15764  063424  000000                    000000                  ;DEST OP1
15765  063426  000000                    000000                  ;ANS1
15766  063430  177777                    177777                  ;SRC OP2
15767  063432  177777                    177777                  ;DST OP2
15768  063434  177777                    177777                  ;ANS2
15769  063436  000000                    000000                  ;SRC OP3
15770  063440  177777                    177777                  ;DST OP3
15771  063442  177777                    177777                  ;ANS3
15772  063444  177777                    177777                  ;SRC OP4
15773  063446  000000                    000000                  ;DST OP4
15774  063450  177777                    177777                  ;ANS4
15775  063452  125252                    125252                  ;SRC OP5
15776  063454  125252                    125252                  ;DST OP5
15777  063456  125252                    125252                  ;ANS5
15778  063460  052525                    052525                  ;SRC OP6
15779  063462  052525                    052525                  ;DST OP6
15780  063464  052525                    052525                  ;ANS6
15781  063466  125252                    125252                  ;SRC OP7
15782  063470  052525                    052525                  ;DST OP7
15783  063472  177777                    177777                  ;ANS7
15784  063474  052525                    052525                  ;SRC OP8
15785  063476  125252                    125252                  ;DST OP8
15786  063500  177777          ALUSUB:   177777                  ;ANS8 -- ALSO NULL ENTRY FOR ALUSUB
15787
15788                          ;THIS TABLE OF 8 ENTRIES IS USED BYTHE ALU SUB TEST IN THE
15789                          ;COMBINED INSTRUCTION EXERCISER TESTS
15790
15791  063502  000000                    000000                  ;SRC OP1
15792  063504  000000                    000000                  ;DST OP1
15793  063506  000000                    000000                  ;ANS1
15794  063510  177777                    177777                  ;SRC OP2
```

```
15795  063512  177777                    177777                  ;DST OP2
15796  063514  000000                    000000                  ;ANS2
15797  063516  125252                    125252                  ;SRC OP3
15798  063520  052525                    052525                  ;DST OP3
15799  063522  125253                    125253                  ;ANS3
15800  063524  052525                    052525                  ;SRC OP4
15801  063526  125252                    125252                  ;DST OP4
15802  063530  052525                    052525                  ;ANS4
15803  063532  125253                    125253                  ;SRC OP5
15804  063534  125252                    125252                  ;DST OP5
15805  063536  000000                    000000                  ;ANS5
15806  063540  052525                    052525                  ;SRC OP6
15807  063542  052525                    052525                  ;DST OP6
15808  063544  000000                    000000                  ;ANS6
15809  063546  052525                    052525                  ;SRC OP7
15810  063550  125253                    125253                  ;DST OP7
15811  063552  052526                    052526                  ;ANS7
15812  063554  125253                    125253                  ;SRC OP8
15813  063556  052525                    052525                  ;DST OP8
15814  063560  125252                    125252                  ;ANS8
15815
15816  063562  005702          INSTAB:   TST     R2              ;BEGINNING OF INSTRUCTION TABLE OF INSTRUCTIONS
15817  063564  005002                    CLR     R2              ;THAT  TEST BUT SERVICE IN VARIOUS ROM LOCATIONS
15818  063566  005102                    COM     R2
15819  063570  005202                    INC     R2
15820  063572  005302                    DEC     R2
15821  063574  005502                    ADC     R2
15822  063576  005602                    SBC     R2
15823  063600  006202                    ASR     R2
15824  063602  006302                    ASL     R2
15825  063604  105002                    CLRB    R2
15826  063606  105102                    COMB    R2
15827  063610  105202                    INCB    R2
15828  063612  105302                    DECB    R2
15829  063614  105502                    ADCB    R2
15830  063616  105502                    ADCB    R2
15831  063620  105602                    SBCB    R2
15832  063622  105702                    TSTB    R2
15833  063624  106202                    ASRB    R2
15834  063626  106302                    ASLB    R2
15835  063630  151302                    BISB    (R3),R2
15836  063632  074302                    XOR     R3,R2
15837  063634  121302                    CMPB    (R3),R2
15838  063636  131302                    BITB    (R3),R2
15839  063640  141302                    BICB    (R3),R2
15840  063642  111302                    MOVB    (R3),R2
15841  063644  021302                    CMP     (R3),R2
15842  063646  031302                    BIT     (R3),R2
15843  063650  041302                    BIC     (R3),R2
15844  063652  051302                    BIS     (R3),R2
15845  063654  006702                    SXT     R2
15846  063656  005402                    NEG     R2
15847  063660  161302                    SUB     (R3),R2
15848  063662  020312                    CMP     R3,(R2)
15849  063664  030312                    BIT     R3,(R2)
15850  063666  120312                    CMPB    R3,(R2)
```

```
15851  063670  131302                          BITB    (R3),R2
15852  063672  005712                          TST     (R2)
15853  063674  105712                          TSTB    (R2)
15854  063676  021312                          CMP     (R3),(R2)
15855  063700  031312                          BIT     (R3),(R2)
15856  063702  121312                          CMPB    (R3),(R2)
15857  063704  131312                          BITB    (R3),(R2)
15858  063706  061302                          ADD     (R3),R2
15859  063710  000302                          SWAB    R2
15860  063712  160302                          SUB     R3,R2
15861  063714  060302                          ADD     R3,R2
15862  063716  010302                          MOV     R3,R2
15863  063720  011302                          MOV     (R3),R2
15864  063722  110302                          MOVB    R3,R2
15865  063724  006102                          ROL     R2
15866  063726  106102                          ROLB    R2
15867  063730  105402                          NEGB    R2
15868  063732  102400                          BVS     .+2
15869  063734  103000                          BVC     .+2
15870  063736  000005                          RESET
15871  063740  020302                          CMP     R3,R2
15872  063742  030302                          BIT     R3,R2
15873  063744  040302                          BIC     R3,R2
15874  063746  120302                          CMPB    R3,R2
15875  063750  130302                          BITB    R3,R2
15876  063752  140302                          BICB    R3,R2
15877  063754  150302                          BISB    R3,R2
15878
15879  063756  000000                  DWTB:   0                       ;ALSO SERVES AS INSTAB TABLE TERMINATOR
15880  063760  000001                          1
15881  063762  000400                          400
15882  063764  177401                          177401
15883  063766  052526                          52526
15884  063770  125253                          125253
15885                                  ;*      MED TEST TABLES
15886
15887  063772  000000                  TLOC1:  .WORD   0
15888  063774  000000                  PSWHOL: .WORD   0
15889  063776  000000                  TABBEG: .WORD   0
15890  064000  000000                  TABEND: .WORD   0
15891  064002  000040                  STGBLK: .BLKW   40
15892  064102  000000                  VADR:   .WORD   0
15893  064104  000000                  PA1716: .WORD   0
15894  064106  000000                  PA1500: .WORD   0
15895  064110  000000                  TLOC2:  .WORD   0
15896
15897                                  ;*
15898                                  ;*      TABLE II
15899                                  ;*
15900                                  ;*      FOLLOWING IS A TABLE OF INTERNAL REGISTER OPERATION CODES
15901                                  ;*      USED FOR TESTING THE MED INSTRUCTION.  LABELS CORRESPOND
15902                                  ;*      TO REGISTER NAMES, THE HIGH BYTE IS THE READ OPERATION
15903                                  ;*      CODE, THE LOW BYTE THE WRITE CODE.
15904                                  ;*              NOTE:  WHEN ADDING OR DELETING
15905                                  ;*              ENTRIES IN THIS TABLE, CHECK DUAL
15906                                  ;*              ADDRESING TEST TO SEE THAT THE "SCRATCH
```

```
15907                                   ;*              PAD LIMITS" ARE MAINTAINED.
15908                                   ;*
15909
15910  064112                           TBL2:
15911
15912  064112                           ASP1:                                   ;A SCRATCH PAD - LO
15913  064112   201      001            R1A:     .BYTE   201,001                ;LOBYTE, HIBYTE=WRITE CODE, READ CODE
15914  064114   202      002            R2A:     .BYTE   202,002
15915  064116   203      003            R3A:     .BYTE   203,003
15916  064120   204      004            R4A:     .BYTE   204,004
15917  064122   205      005            R5A:     .BYTE   205,005
15918  064124   206      006            R6A:     .BYTE   206,006
15919  064126   210      010            FAC3.0:  .BYTE   210,010
15920  064130   211      011            FAC3.1:  .BYTE   211,011
15921  064132   212      012            FAC3.2:  .BYTE   212,012
15922  064134   213      013            FAC3.3:  .BYTE   213,013
15923  064136   214      014            FAC3.4:  .BYTE   214,014
15924  064140   215      015            FAC3.5:  .BYTE   215,015
15925  064142   216      016            UR6A:    .BYTE   216,016
15926  064144   217      017            FDST3:   .BYTE   217,017
15927  064146   220      020            WCSA.0:  .BYTE   220,020                ;A SCRATCH PAD-HI
15928  064150   221      021            WCSA.1:  .BYTE   221,021
15929  064152   222      022            GNWHAM:  .BYTE   222,022
15930  064154   223      023            CNSTSW:  .BYTE   223,023
15931  064156   226      026            CNSSW:   .BYTE   226,026
15932  064160   227      027            CNSCDR:  .BYTE   227,027
15933  064162   230      030            FAC1.0:  .BYTE   230,030
15934  064164   231      031            FAC1.1:  .BYTE   231,031
15935  064166   232      032            FAC1.2:  .BYTE   232,032
15936  064170   233      033            FAC1.3:  .BYTE   233,033
15937  064172   234      034            FAC1.4:  .BYTE   234,034
15938  064174   235      035            FAC1.5:  .BYTE   235,035
15939  064176   236      036            FPSHI:   .BYTE   236,036
15940  064200   237      037            ASP2:    FDST1:  .BYTE   237,037
15941
15942  064202                           BSP1:
15943  064202   241      041            R1B:     .BYTE   241,041                ;B SCRATCH PAD - LO
15944  064204   242      042            R2B:     .BYTE   242,042
15945  064206   243      043            R3B:     .BYTE   243,043
15946  064210   244      044            R4B:     .BYTE   244,044
15947  064212   245      045            R5B:     .BYTE   245,045
15948  064214   246      046            R6B:     .BYTE   246,046
15949  064216   250      050            FAC2.0:  .BYTE   250,050
15950  064220   251      051            FAC2.1:  .BYTE   251,051
15951  064222   252      052            FAC2.2:  .BYTE   252,052
15952  064224   253      053            FAC2.3:  .BYTE   253,053
15953  064226   254      054            FAC2.4:  .BYTE   254,054
15954  064230   255      055            FAC2.5:  .BYTE   255,055
15955  064232   256      056            UR6B:    .BYTE   256,056
15956  064234   257      057            FDST2:   .BYTE   257,057
15957  064236   260      060            WCSB.0:  .BYTE   260,060                ;B SCRATCH PAD - HI
15958  064240   261      061            WCSB.1:  .BYTE   261,061
15959  064242   262      062            WCSADR:  .BYTE   262,062
15960  064244   263      063            RZERO:   .BYTE   263,063
15961  064246   266      066            RVECT:   .BYTE   266,066
15962  064250   270      070            FAC0.0:  .BYTE   270,070
```

```
15963  064252      272     072          FACO.1: .BYTE   272,072
15964  064254      273     073          FACO.2: .BYTE   273,073
15965  064256      274     074          FACO.4: .BYTE   274,074
15966  064260      275     075          FACO.5: .BYTE   275,075
15967  064262      276     076          FEA:    .BYTE   276,076
15968  064264      277     077          BSP2:   FDSTO:  .BYTE   277,077
15969
15970  064266                           CSP1:
15971  064266      300     100          LJAM:   .BYTE   300,100         ;C SCRATCH PAD
15972  064270      301     101          LSERV:  .BYTE   301,101
15973  064272      302     102          LPBA:   .BYTE   302,102
15974  064274      303     103          LCUA:   .BYTE   303,103
15975  064276      304     104          LFGIN:  .BYTE   304,104
15976  064300      305     105          LWHAM:  .BYTE   305,105
15977  064302      307     107          LTAG:   .BYTE   307,107
15978  064304      310     110          CNSC0:  .BYTE   310,110
15979  064306      311     111          CNSC1:  .BYTE   311,111
15980  064310      312     112          CNSC2:  .BYTE   312,112
15981  064312      313     113          CST200: .BYTE   313,113
15982  064314      316     116          CSP2:   CNSTO:  .BYTE   316,116
15983  064316      224     024          RT1A:   .BYTE   224,024
15984  064320      225     025          RT2A:   .BYTE   225,025
15985  064322      264     064          RT1B:   .BYTE   264,064
15986  064324      265     065          RT2B:   .BYTE   265,065
15987  064326   000000                          .WORD   0
15988
15989                                    ;*
15990                                    ;*      TABLE III
15991                                    ;*
15992                                    ;*      THE FOLLOWING IS A LIST OF "NOP" OPERATION CODES
15993                                    ;*      THAT WILL BE USED WITH A MED IN MED TEST 3 TO
15994                                    ;*      ENSURE THAT A MED WITH THESE CODES WILL NOT HANG.
15995
15996  064330                           TBL3:
15997  064330      120     137          NOPS:   .BYTE   120,137         ;GROUP A
15998  064332      145     145                  .BYTE   145,145         ;GROUP B
15999  064334      150     151                  .BYTE   150,151         ;GROUP C
16000  064336      156     177                  .BYTE   156,177         ;GROUP D
16001  064340      320     343                  .BYTE   320,343         ;GROUP E
16002  064342      353     357                  .BYTE   353,357         ;FROUP G
16003  064344   000000                          .WORD   0               ;A 0 TERMINATES TABLE
16004
16005                                    ;*
16006                                    ;*      TABLE IV
16007                                    ;*
16008                                    ;*      THE LIST BELOW CONTAINS THOSE OPERATION CODES
16009                                    ;*      CORRESPONDING TO THE INTERNAL REGISTERS WHICH MUST
16010                                    ;*      BE TESTED SEPERATELY BECAUSE THEY ARE READ-ONLY,
16011                                    ;*      WRITE-ONLY, OR USED IN MACRO CODE EXECUTION, ETC. . .
16012
16013  064346                           TBL4:
16014  064346      200     000          R0A:    .BYTE   200,000         ;LOBYTE,HYBYTE - WRITE CODE, READ CODE
16015  064350      207     007          R7A:    .BYTE   207,007         ;0 REPLACES ANY NON EXSISTENT CODES
16016  064352      240     040          R0B:    .BYTE   240,040         ;EXCEPT IN THE CASE OF R0A
16017  064354      247     047          R7B:    .BYTE   247,047
16018  064356      314     114          CNST2:  .BYTE   314,114
```

```
16019  064360  317  117          CNST1:  .BYTE   317,117
16020                            ;*      TABLE V
16021                            ;*
16022  064362                    TBL5:
16023
16024  064362  306               LCDTA:  .BYTE   306          ;THIS TABLE CONTAINS THE OPERATION
16025  064363  106                       .BYTE   106          ;CODES OF THOSE INTERNAL REGISTERS
16026  064364  315               MD:     .BYTE   315          ;WHICH MUST BE TESTED USING THE
16027  064365  115                       .BYTE   115          ;MICROBREAK REGISTER.  THEIR
16028  064366  267               CNSCTL: .BYTE   267          ;ASSOCIATED MICRO-ADDRESSES ARE IN
16029  064367  067                       .BYTE   067          ;THE NEXT TABLE
16030  064370  140               JAM:    .BYTE   140
16031  064371  141               SERV:   .BYTE   141
16032  064372  142               PBA:    .BYTE   142
16033  064373  143               CUA:    .BYTE   143
16034  064374  344               FLAG:   .BYTE   344
16035  064375  144                       .BYTE   144
16036  064376  345               DREG:   .BYTE   345
16037  064377  146               REV:    .BYTE   146
16038  064400  346               SREG:   .BYTE   346
16039  064401  147               COUNT:  .BYTE   147
16040  064402  347               NUA:    .BYTE   347
16041  064403  351               RES:    .BYTE   351
16042  064404  152               DCS0:   .BYTE   152
16043  064405  352                       .BYTE   352          ;INIT REG
16044  064406  153               DCS1:   .BYTE   153
16045  064407  000                       .BYTE   0            ;TABLE TERMINATOR
16046                                    .EVEN
16047
16048                            ;*      TABLE VI
16049                            ;*
16050  064410                    TBL6:
16051
16052  064410  003330            ULCDTA: .WORD   3330         ;THIS TABLE CONTAINS THE MICRO-ADDRESSES
16053  064412  003150                    .WORD   3150         ;WHICH ARE LOADED INTO THE MICROBREAK
16054  064414  003375            UMD:    .WORD   3375         ;REG. TO TEST THE OPERATION CODES
16055  064416  003271                    .WORD   3271         ;CONTAINED IN THE PRECEEDING TABLE.
16056  064420  003240            UCNSCTL:.WORD   3240
16057  064422  003224                    .WORD   3224
16058  064424  003160            UJAM:   .WORD   3160
16059  064426  003161            USERV:  .WORD   3161
16060  064430  003170            UPBA:   .WORD   3170
16061  064432  003171            UCUA:   .WORD   3171
16062  064434  003344            UFLAG:  .WORD   3344
16063  064436  003320                    .WORD   3320
16064  064440  003345            UDREG:  .WORD   3345
16065  064442  003340            UREV:   .WORD   3340
16066  064444  003350            USREG:  .WORD   3350
16067  064446  003341            UCOUNT: .WORD   3341
16068  064450  003351            UNUA:   .WORD   3351
16069  064452  003355            URES:   .WORD   3355
16070  064454  003720            UDCS0:  .WORD   3720
16071  064456  003724            UINIT:  .WORD   3724
16072  064460  003721            UDCS1:  .WORD   3721
16073
16074                            ;*      TABLE VII
```

```
16075                                   ;*
16076                                   ;*      THIS TABLE HOLDS THE OPERATION CODES AND THE CONSTANT
16077                                   ;*      VALUE EXPECTED FOR CERTAIN INTERNAL REGISTERS.
16078  064462                           TBL7:
16079
16080  064462  000100  077600          CLJAM:  .WORD   100,77600
16081  064466  000101  000010          CLSERV: .WORD   101,10
16082  064472  000102  020000          CLPBA:  .WORD   102,20000
16083  064476  000103  000004          CLCUA:  .WORD   103,4
16084  064502  000104  050000          CLFGIN: .WORD   104,50000
16085  064506  000105  054000          CLWHAM: .WORD   105,54000
16086  064512  000107  024000          CLTAG:  .WORD   107,24000
16087  064516  000110  177400          CCNSC0: .WORD   110,177400
16088  064522  000111  177600          CCNSC1: .WORD   111,177600
16089  064526  000112  100000          CCNSC2: .WORD   112,100000
16090  064532  000113  000200          CCST200:.WORD   113,200
16091  064536  000114  000002          CCNST2: .WORD   114,2
16092  064542  000116  000000          CCNST0: .WORD   116,0
16093  064546  000117  000001          CCNST1: .WORD   117,1
16094  064552  000000                          .WORD   0
16095
16096                                   .EVEN
16097  064554                           DBTA:
16098  064554     000     377     252  .BYTE   000,377,252,125
16099  064557     125
16100  064560                           DBTB:
16101  064560     000     001     120  .BYTE   000,001,120,253
16102  064563     253
16103
16104                                   ;MESSAGE TABLES
16105
16106  064564                           EM1:
16107  064564                           EM2:
16108  064564  027523  020102  051504  EM4:    .ASCII  'S/B DST '
16109  064572  020124
16110  064574  040527  020123  051504  EM7:    .ASCII  'WAS DST '
16111  064602  020124
16112  064604  042040  051505  004524  EM6:    .ASCII  ' DEST'<HT>
16113  064612  024040  051111  004451  EM5:    .ASCIZ  ' (IR)'<HT>' TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16114  064620  052040  051505  004524
16115  064626  024040  041520  004451
16116  064634  024040  050123  004451
16117  064642  050050  053523  000051
16118  064650  027523  020102  042522  EM10:   .ASCIZ  'S/B RES WAS RES DST OP  SRC OP  TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16119  064656  020123  040527  020123
16120  064664  042522  020123  051504
16121  064672  020124  050117  020040
16122  064700  051123  020103  050117
16123  064706  020040  042524  052123
16124  064714  020011  050050  024503
16125  064722  020011  051450  024520
16126  064730  024011  051520  024527
16127  064736     000
16128  064737     123  041057  051440  EM3:    .ASCIZ  'S/B SP'<HT>'WAS SP'<HT>' (IR)'<HT>' TEST'<HT>' (PC)'<HT>'(PSW)'
16129  064744  004520  040527  020123
16130  064752  050123  020011  044450
```

```
16131  064760  024522  020011  042524
16132  064766  052123  020011  050050
16133  064774  024503  024011  051520
16134  065002  024527     000
16135  065005     011  020011  051511  DH2:    .ASCIZ  <HT><HT>' IS R3'
16136  065012  051040  000063
16137  065016  004411  044440  020123  DH4:    .ASCIZ  <HT><HT>' IS R5'
16138  065024  032522     000
16139  065027     015  042412  042116  EOP1:   .ASCIZ  <15><12>'END PASS # '
16140  065034  050040  051501  020123
16141  065042  020043     000
16142  065045     011  051105  047522  EOP2:   .ASCIZ <HT>'ERROR COUNT = '
16143  065052  020122  047503  047125
16144  065060  020124  020075     000
16145  065065     015  046412  026504  IDENT1: .ASCIZ <15><12>'MD-11-DQKDA-A   KD11-K BASIC LOGIC TESTS'<15><12>
16146  065072  030461  042055  045521
16147  065100  040504  040455  020040
16148  065106  045440  030504  026461
16149  065114  020113  040502  044523
16150  065122  020103  047514  044507
16151  065130  020103  042524  052123
16152  065136  006523  000012
16153  065142  005015  051124  050101  BEMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 4 PC = '
16154  065150  042520  020104  047524
16155  065156  032040  050040  020103
16156  065164  020075     000
16157  065167     015  052012  040522  RSMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 10 PC = '
16158  065174  050120  042105  052040
16159  065202  020117  030061  050040
16160  065210  020103  020075     000
16161  065215     124  051505  051524  EM11:   .ASCIZ  'TESTS SKIPPED'
16162  065222  051440  044513  050120
16163  065230  042105     000
16164  065233     040  050040  004503  DH11:   .ASCIZ  "  PC"<HT>"EXPCTD"<HT>"ACTUAL"<HT>"(TEST #'S)"
16165  065240  054105  041520  042124
16166  065246  040411  052103  040525
16167  065254  004514  052050  051505
16168  065262  020124  023443  024523
16169  065270     000
16170  065271     115  042105  042040  EM12:   .ASCIZ  /MED DID NOT ABORT IN USER MODE/
16171  065276  042111  047040  052117
16172  065304  040440  047502  052122
16173  065312  044440  020116  051525
16174  065320  051105  046440  042117
16175  065326  000105
16176  065330  042515  020104  054105  EM13:   .ASCIZ  /MED EXECUTED IN USER MODE/
16177  065336  041505  052125  042105
16178  065344  044440  020116  051525
16179  065352  051105  046440  042117
16180  065360  000105
16181  065362  042515  020104  044103  EM14:   .ASCIZ  /MED CHANGED PSW/
16182  065370  047101  042507  020104
16183  065376  051520  000127
16184  065402  044515  051103  041117  EM15:   .ASCIZ  /MICROBREAK TRAP-TO-4 DID NOT OCCUR/
16185  065410  042522  045501  052040
16186  065416  040522  026520  047524
```

```
16187  065424  032055  042040  042111
16188  065432  047040  052117  047440
16189  065440  041503  051125     000
16190  065445     114  043517  052503  EM17:   .ASCIZ  /LOGCUA LOGGED WRONG/
16191  065452  020101  047514  043507
16192  065460  042105  053440  047522
16193  065466  043516     000
16194  065471     103  050123  041440  EM21:   .ASCIZ  /CSP CONSTANT WRONG/
16195  065476  047117  052123  047101
16196  065504  020124  051127  047117
16197  065512  000107
16198  065514  040502  020104  040504  EM22:   .ASCIZ  /BAD DATA READ BY A MED/
16199  065522  040524  051040  040505
16200  065530  020104  054502  040440
16201  065536  046440  042105     000
16202  065543     116  020117  042117  EM23:   .ASCIZ  /NO ODD PC TRAP/
16203  065550  020104  041520  052040
16204  065556  040522  000120
16205  065562  042117  020104  042101  EM24:   .ASCIZ  /ODD ADR. BIT NOT SET IN CPU ERR REG OR LOG JAM/
16206  065570  027122  041040  052111
16207  065576  047040  052117  051440
16208  065604  052105  044440  020116
16209  065612  050103  020125  051105
16210  065620  020122  042522  020107
16211  065626  051117  046040  043517
16212  065634  045040  046501     000
16213  065641     120  054510  020123  EM26:   .ASCIZ  /PHYS BA LOGGED WRONG/
16214  065646  040502  046040  043517
16215  065654  042507  020104  051127
16216  065662  047117  000107
16217  065666  040503  044103  020105  EM27:   .ASCIZ  /CACHE PARITY ERROR LOGGED IN BAKUP MODE/
16218  065674  040520  044522  054524
16219  065702  042440  051122  051117
16220  065710  046040  043517  042507
16221  065716  020104  047111  041040
16222  065724  045501  050125  046440
16223  065732  042117  000105
16224  065736  040503  044103  020105  EM30:   .ASCIZ  /CACHE PARITY TRAPPED WHEN DISABLED/
16225  065744  040520  044522  054524
16226  065752  052040  040522  050120
16227  065760  042105  053440  042510
16228  065766  020116  044504  040523
16229  065774  046102  042105     000
16230  066001     111  051516  051124  EM41:   .ASCIZ  /INSTR. NOT ABORTED IN CACHE ABORT MODE/
16231  066006  020056  047516  020124
16232  066014  041101  051117  042524
16233  066022  020104  047111  041440
16234  066030  041501  042510  040440
16235  066036  047502  052122  046440
16236  066044  042117  000105
16237  066050  042515  047515  054522  EM32:   .ASCIZ  /MEMORY ERR REG INCORRECT/
16238  066056  042440  051122  051040
16239  066064  043505  044440  041516
16240  066072  051117  042522  052103
16241  066100     000
16242  066101     124  046511  047505  EM33:   .ASCIZ  /TIMEOUT BIT NOT SET IN CPU ERR REG OR LOG JAM/
```

```
 16243  066106  052125  041040  052111
 16244  066114  047040  052117  051440
 16245  066122  052105  044440  020116
 16246  066130  050103  020125  051105
 16247  066136  020122  042522  020107
 16248  066144  051117  046040  043517
 16249  066152  045040  046501     000
 16250  066157     116  020117  046111  EM34:   .ASCIZ  /NO ILLEGAL INTERNAL ADR TRAP/
 16251  066164  042514  040507  020114
 16252  066172  047111  042524  047122
 16253  066200  046101  040440  051104
 16254  066206  052040  040522  000120
 16255  066214  047111  051124  040516  EM35:   .ASCIZ  /INTRNAL ADR ERR BIT NOT SET IN CPU ERR REG OR LOG JAM/
 16256  066222  020114  042101  020122
 16257  066230  051105  020122  044502
 16258  066236  020124  047516  020124
 16259  066244  042523  020124  047111
 16260  066252  041440  052520  042440
 16261  066260  051122  051040  043505
 16262  066266  047440  020122  047514
 16263  066274  020107  040512  000115
 16264  066302  040514  052123  044440  EM36:   .ASCIZ  "LAST INTR/TRAP VECTOR NOT LOGGED IN FLAG REG"
 16265  066310  052116  027522  051124
 16266  066316  050101  053040  041505
 16267  066324  047524  020122  047516
 16268  066332  020124  047514  043507
 16269  066340  042105  044440  020116
 16270  066346  046106  043501  051040
 16271  066354  043505     000
 16272  066357     114  043517  043040  EM37:   .ASCIZ  /LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR/
 16273  066364  051111  052123  046440
 16274  066372  042117  020105  044504
 16275  066400  020104  047516  020124
 16276  066406  047111  044510  044502
 16277  066414  020124  051105  047522
 16278  066422  020122  047514  020107
 16279  066430  043101  042524  020122
 16280  066436  044506  051522  020124
 16281  066444  051105  047522  000122
 16282  066452  051105  047522  020122  EM40:   .ASCIZ  /ERROR LOG WAS NOT REENABLED, ODD ADR BIT CLR IN CPUERR/
 16283  066460  047514  020107  040527
 16284  066466  020123  047516  020124
 16285  066474  042522  047105  041101
 16286  066502  042514  026104  047440
 16287  066510  042104  040440  051104
 16288  066516  041040  052111  041440
 16289  066524  051114  044440  020116
 16290  066532  050103  042525  051122
 16291  066540     000
 16292  066541     116  020117  040503  EM31:   .ASCIZ  /NO CACHE PARITY TRAP/
 16293  066546  044103  020105  040520
 16294  066554  044522  054524  052040
 16295  066562  040522  000120
 16296  066566  047514  023040  044040  EM42:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE/
 16297  066574  020111  054502  042524
 16298  066602  023040  052040  043501
```

```
16299  066610  050040  051101  052111
16300  066616  020131  044502  051524
16301  066624  047040  052117  051440
16302  066632  052105  044440  020116
16303  066640  047514  020107  042523
16304  066646  053122  041511  000105
16305  066654  047514  023040  044040  EM43:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN MEM ERR REG/
16306  066662  020111  054502  042524
16307  066670  023040  052040  043501
16308  066676  050040  051101  052111
16309  066704  020131  044502  051524
16310  066712  047040  052117  051440
16311  066720  052105  044440  020116
16312  066726  042515  020115  051105
16313  066734  020122  042522  000107
16314  066742  040503  044103  020105  EM45:   .ASCIZ  /CACHE TAG LOGGED WRONG/
16315  066750  040524  020107  047514
16316  066756  043507  042105  053440
16317  066764  047522  043516     000
16318  066771     103  041501  042510  EM16:   .ASCIZ  /CACHE DATA LOGGED WRONG/
16319  066776  042040  052101  020101
16320  067004  047514  043507  042105
16321  067012  053440  047522  043516
16322  067020     000
16323  067021     105  051511  051440  EMEIS1: .ASCIZ  'EIS SET COND CODES WRONG'
16324  067026  052105  041440  047117
16325  067034  020104  047503  042504
16326  067042  020123  051127  047117
16327  067050  000107
16328  067052  044505  020123  040507  EMEIS2: .ASCIZ  'EIS GAVE WRONG RESULT'
16329  067060  042526  053440  047522
16330  067066  043516  051040  051505
16331  067074  046125  000124
16332  067100  052501  047524  044455  EM46:   .ASCIZ  'AUTO-INCREMENT (DECREMT) DID NOT OCCUR IN EIS'
16333  067106  041516  042522  042515
16334  067114  052116  024040  042504
16335  067122  051103  046505  024524
16336  067130  042040  042111  047040
16337  067136  052117  047440  041503
16338  067144  051125  044440  020116
16339  067152  044505  000123
16340  067156  050040  053523  051011  DHEIS1: .ASCII  ' PSW'<HT>'REG-WAS-REG+1'<HT>'REG-S/B-REG+1'<HT>
16341  067164  043505  053455  051501
16342  067172  051055  043505  030453
16343  067200  051011  043505  051455
16344  067206  041057  051055  043505
16345  067214  030453     011
16346  067217     040  050040  004503  DH46:   .ASCIZ  '  PC'<HT>' (IR)'<HT>' TEST'
16347  067224  024040  051111  004451
16348  067232  052040  051505  000124
16349  067240  020040  041520  046411  DH15:   .ASCIZ  /  PC/<HT>/MEDCODE MICROBK REG./
16350  067246  042105  047503  042504
16351  067254  046440  041511  047522
16352  067262  045502  051040  043505
16353  067270  000056
16354  067272  020040  041520  046411  DH17:   .ASCIZ  /  PC/<HT>/MEDCODE EXPECTD RECEIVD/
```

```
 16355  067300  042105  047503  042504
 16356  067306  042440  050130  041505
 16357  067314  042124  051040  041505
 16358  067322  044505  042126     000
 16359  067327     040  050040  000103  DH23:   .ASCIZ  /  PC/
 16360  067334  020040  041520  041411  DH24:   .ASCIZ  /  PC/<HT>/CPUERR/<HT>/LOGJAM/
 16361  067342  052520  051105  004522
 16362  067350  047514  045107  046501
 16363  067356     000
 16364  067357     040  050040  004503  DH25:   .ASCIZ  /  PC/<HT>/FLGREG/
 16365  067364  046106  051107  043505
 16366  067372     000
 16367  067373     040  050040  004503  DH26:   .ASCIZ  '  PC'<HT>'<17:16>-S/B PA-<15:0>  <17:16>-WAS PA-<15:0>'
 16368  067400  030474  035067  033061
 16369  067406  026476  027523  020102
 16370  067414  040520  036055  032461
 16371  067422  030072  020076  036040
 16372  067430  033461  030472  037066
 16373  067436  053455  051501  050040
 16374  067444  026501  030474  035065
 16375  067452  037060     000
 16376  067455     040  050040  004503  DH27:   .ASCIZ  /  PC/<HT>/LOGPBA/<HT>/LOGDATA/<HT>/LOGTAG/
 16377  067462  047514  050107  040502
 16378  067470  046011  043517  040504
 16379  067476  040524  046011  043517
 16380  067504  040524  000107
 16381  067510  020040  041520  046411  DH32:   .ASCIZ  /  PC/<HT>/MEMERR/
 16382  067516  046505  051105  000122
 16383  067524  020040  041520  046011  DH42:   .ASCIZ  /  PC/<HT>/LOGSERVCE/
 16384  067532  043517  042523  053122
 16385  067540  042503     000
 16386  067543     040  050040  004503  DH44:   .ASCIZ  /  PC/<HT>/EXPCT/<HT>/RECVD/
 16387  067550  054105  041520  004524
 16388  067556  042522  053103  000104
 16389                          .EVEN
 16390  067564  001016  001076  001100  DT15:   .WORD   $ERRPC,$TMP0,$TMP1,0
 16391  067572  000000
 16392  067574  001016  001100  001102  DT21:   .WORD   $ERRPC,$TMP1,$TMP2,$REG0,0
 16393  067602  001062  000000
 16394  067606  001016  001100  001102  DT22:   .WORD   $ERRPC,$TMP1,$TMP2,$TMP3,0
 16395  067614  001104  000000
 16396  067620  001016  000000          DT23:   .WORD   $ERRPC,0
 16397  067624  001016  001064  001062  DT24:   .WORD   $ERRPC,$REG1,$REG0,0
 16398  067632  000000
 16399  067634  001016  001062  000000  DT25:   .WORD   $ERRPC,$REG0,0
 16400  067642  001016  001064  001066  DT26:   .WORD   $ERRPC,$REG1,$REG2,$REG0,$REG3,0
 16401  067650  001062  001070  000000
 16402  067656  001016  001070  001064  DT27:   .WORD   $ERRPC,$REG3,$REG1,$REG2,0
 16403  067664  001062  000000
 16404  067670  001060  001066  001070  DTEIS1: .WORD   $REGAD,$REG2,$REG3,$REG1,$REG4
 16405  067676  001064  001072
 16406  067702  001016  001076  001062  DT46:   .WORD   $ERRPC,$TMP0,$REG0,0
 16407  067710  000000
 16408
 16409  067712     000     000          DF15:   .BYTE   0,0
 16410  067714     000     000     000  DF17:   .BYTE   0,0,0
```

```
16411                 067720                        .EVEN
16412  067720                                       DT1:
16413  067720                                       DT2:
16414  067720                                       DT4:
16415  067720  001072                               DT10:   .WORD   $REG4
16416  067722  001070                               DT7:    .WORD   $REG3
16417  067724  001066                               DT6:    .WORD   $REG2
16418  067726  001064  001062  001016               DT5:    .WORD   $REG1,$REG0,$ERRPC,$REG5,$REGAD,0
16419  067734  001074  001060  000000
16420  067742  001072  001070  001064               DT3:    .WORD   $REG4,$REG3,$REG1,$REG0,$ERRPC,$REGAD,0
16421  067750  001062  001016  001060
16422  067756  000000
16423  067760  001016  001124  001062               DT11:   .WORD   $ERRPC,$TESTN,$REG0,0
16424  067766  000000
16425                 000001                        .END
```

```
ABASE = 000000            237
ACDW1 = 000000            237
ACDW2 = 000000            237
ACPUOP= 000000            237     252
ADDW0 = 000000            237
ADDW1 = 000000            237
ADDW10= 000000            237
ADDW11= 000000            237
ADDW12= 000000            237
ADDW13= 000000            237
ADDW14= 000000            237
ADDW15= 000000            237
ADDW2 = 000000            237
ADDW3 = 000000            237
ADDW4 = 000000            237
ADDW5 = 000000            237
ADDW6 = 000000            237
ADDW7 = 000000            237
ADDW8 = 000000            237
ADDW9 = 000000            237
ADEVCT= 000000            237     243
ADEVM = 000000            237
AENV  = 000000            237     248
AENVM = 000000            237     249
AFATAL= 000000            237     240
ALUADD  063260          13302   13307   15702#
ALUSUB  063500          13362   13367   15786#
AMADR1= 000000            237
AMADR2= 000000            237
AMADR3= 000000            237
AMADR4= 000000            237
AMAMS1= 000000            237
AMAMS2= 000000            237
AMAMS3= 000000            237
AMAMS4= 000000            237
AMSGAD= 000000            237     245
AMSGLG= 000000            237     246
AMSGTY= 000000            237     239
AMTYP1= 000000            237
AMTYP2= 000000            237
AMTYP3= 000000            237
AMTYP4= 000000            237
AMDTAB  063340          13422   13427   15730#
APASS = 000000            237     242
APRIOR= 000000            237
APTCSU= 000040          15440   15624#
APTENV= 000001          15329   15433   15580   15622#
APTSIZ= 000200          15621#
APTSP0= 000100          15435   15582   15623#
ASP1    064112          15912#
ASP2    064200          15940#
ASWREG= 000000            237     250
ATA     063222           1938    1988    2000    2016    2048    2061    2109    2130    2149    2161    2171    2183    2194
                         6927    6938    6951    6962    6975    6987    7000    7011    7024    7036    7049    7068    7089
                         7110    7131    7160    7189    7190    7204    7218    7219    7233    7247    7276    7305    7306
                         7320    7334    7335    7349    7363    7385    7407    7429    8067    8079    8119    8131    8166
```

```
                        8391    8405    8428    8442    8543    8557    8580    8594    8681    8713    8785    8827    9062
                        9067    9116    9121    9165    9230   10379   10422   10466   13704   13724   13776   13796   13843
                       15686#
ATESTN= 000000           237     241
AUNIT = 000000           237     244
AUSWR = 000000           237     251
AVECT1= 000000           237
AVECT2= 000000           237
BOOT    001636           589#    593
BO15    002106           755     760#
BO17    002170           797     802#
BO20    002222           820     825#
BO21    002252           842     847#
BO23    002340           883     888#
BO32    002544           988     993#
BO33    002604          1010    1015#
BO34    002644          1031    1036#
BO41    003010          1112    1117#
BO46    003210          1210    1215#
BASIC   003222          1216    1224#
BEMSG   065142         15200   16153#
BERFLG  063174          3763#   3769#  15191#  15673#
BERR    061144          3801   11610   11614   12478   12516   12555   12572   12624   12684   12722   12753   12785   14212
                       14434   14460   14626   14656   14704   14832   14879   14910   14926   14945   14960   15194#
BETST   061136          3760   15191#
BIT0  = 000001           119#    526    1594   14451   14708   14745   14826   14863   14900   14927
BIT00 = 000001           109#    119
BIT01 = 000002           108#    118
BIT02 = 000004           107#    117
BIT03 = 000010           106#    116
BIT04 = 000020           105#    115
BIT05 = 000040           104#    114
BIT06 = 000100           103#    113
BIT07 = 000200           102#    112
BIT08 = 000400           101#    111
BIT09 = 001000           100#    110   15254   15339
BIT1  = 000002           118#   2255   14776
BIT10 = 002000            99#   9321
BIT11 = 004000            98#  10191   15261
BIT12 = 010000            97#  11003
BIT13 = 020000            96#  12057   14637   15324
BIT14 = 040000            95#  13197   15236
BIT15 = 100000            94#  14148   14382   14409   14451   14470   14745   14786   14863   14893   14900   15138
BIT2  = 000004           117#   2987   14470   14668   14893
BIT3  = 000010           116#   3496
BIT4  = 000020           115#   4219   14630   14883
BIT5  = 000040           114#    530    5092   14714
BIT6  = 000100           113#    525     528    5944   14464   14660   14911
BIT7  = 000200           112#    527     529    6800   14637   14813
BIT8  = 000400           111#   7615
BIT9  = 001000           110#   8536   14377   14428   14637
BKFLAG  061036         14374#  14386   14406#  14416   15081#  15084#
BKROUT  061030         14372   15081#
BPTLOC  063160          1594    2255    2987    3496    4219    5092    5944    6800    7615    8536    9321   10191   11003
                       12057   13197   14148   15666#
BPTVEC= 000014           126#
```

N09


```
BSP1    064202          15942#
BSP2    064264          15968#
BT001   001630            579#
BT002   001634            579     587#
BT003   001646            589     601#    608
BT004   001662            604     614#    623
BT005   001700            619     629#    638
BT006   001716            634     644#    651
BT007   001730            648     657#    666
BT010   001744            662     672#    681
BT011   001762            678     687#    697
BT012   002002            694     703#    712
BT013   002022            709     718#    729
BT014   002046            726     735#    744
BT015   002070            741     750#    758     765
BT016   002116            762     771#    782
BT017   002142            779     788#    800     807
BT020   002202            804     813#    823     829
BT021   002232            826     835#    845     852
BT022   002264            849     858#    868
BT023   002310            865     874#    886     892
BT024   002350            889     898#    907
BT025   002374            904     913#    921
BT026   002414            918     926#    934
BT027   002434            931     940#    948
BT030   002454            945     954#    962
BT031   002474            959     967#    975
BT032   002514            972     981#    991     997
BT033   002554            994    1003#   1013    1019
BT034   002614           1016    1024#   1034    1040
BT035   002654           1037    1045#   1054
BT036   002674           1051    1059#   1068
BT037   002714           1065    1073#   1082
BT040   002734           1079    1088#   1099
BT041   002762           1096    1104#   1115    1121
BT042   003020           1118    1127#   1137
BT043   003040           1134    1143#   1154
BT044   003062           1151    1160#
BT045   003110           1167    1170    1175#   1185
BT046   003136           1182    1191#   1213    1219
CATERR  063176          15126   15128*  15142*  15194   15196*  15203*  15674#
CCNSC0  064516          16087#
CCNSC1  064522          16088#
CCNSC2  064526          16089#
CCNST0  064542          16092#
CCNST1  064546          16093#
CCNST2  064536          16091#
CCR   = 177746            524#  14515*  14517*  14567*  14581*  14583*  14698*  14699   14738*  14742*  14744*  14748*  14763*
CCST20  064532          16090#
CITST   013352           3904    3916#
CLCUA   064476          16083#
CLFGIN  064502          16084#
CLJAM   064462          16080#
CLPBA   064472          16082#
CLSERV  064466          16081#
CLTAG   064512          16086#
```

```
CLHHAM  064506          16085#
CNSCDR  064160          15932#
CNSCTL  064366          16028#
CNSCO   064304          15978#
CNSC1   064306          15979#
CNSC2   064310          15980#
CNSSW   064156          15931#
CNSTSW  064154          15930#
CNST0   064314          15982#
CNST1   064360          16019#
CNST2   064356          16018#
COUNT   064401          16039#
CPUERR= 177766            523#  14463   14629   14659   14707   14882   14911   14919
CR    = 000015             34#  15479   15489   16153   16157
CRLF  = 000200             35#  15450   15489
CSP1    064266          15970#
CSP2    064314          15982#
CST200  064312          15981#
CUA     064373          16033#
DBTA    064554           2067    2613    2625    2663    2675    2863    2871    8924    8944    8963    8974    8987    8998
                         9012    9033    9060    9087    9114    9141    9163   10314   10334   10355   10377   10399   10421
                        10443   10465   10551   15688   16097#
DBTB    064560           2638    2650    2688    2700    2739    2791    2952    2992   15689   16100#
DCS0    064404          16042#
DCS1    064406          16044#
DDISP = 177570             41#     206
DF15    067712            345   16409#
DF17    067714            372     378   16410#
DHEIS1  067156            501     508   16340#
DH11    065233            319   16164#
DH15    067240            343   16349#
DH17    067272            370     376   16354#
DH2     065005            284   16135#
DH23    067327            325     331     337     382     417     424     445     480   16359#
DH24    067334            389     438     452     466     473   16360#
DH25    067357            459   16364#
DH26    067373            403   16367#
DH27    067455            410   16376#
DH32    067510            431     494   16381#
DH4     065016            294   16137#
DH42    067524            487   16383#
DH44    067543            350     357     364     396   16386#
DH46    067217            515   16346#
DISPLA  001042            206#  15276*  15319*
DISPRE  000174            141#
DPTRP = 000001            526#  14517   14738   14748   14763
DREG    064376          16036#
DSWR  = 177570             40#     205
DTEIS1  067670            502     509   16404#
DT1     067720            280   16412#
DT10    067720            315   16415#
DT11    067760            320   16423#
DT15    067564            344   16390#
DT2     067720            285   16413#
DT21    067574            371   16392#
DT22    067606            377   16394#
```

```
DT23    067620          326    332    338    383    418    425    446    481  16396#
DT24    067624          351    358    365    390    397    439    453    467    474  16397#
DT25    067634          432    460    488    495  16399#
DT26    067642          404  16400#
DT27    067656          411  16402#
DT3     067742          290  16420#
DT4     067720          295  16414#
DT46    067702          516  16406#
DT5     067726          300  16418#
DT6     067724          305  16417#
DT7     067722          310  16416#
DWTA    063246         1619   1621   1822   1868   1872   1891   1895   1940   1994   2012   2043   2103   2129
                       2148   2177   2210   2463   2464   2475   2481   2500   2501   2502   2512   2518   2537
                       2538   2539   2549   2555   2574   2575   2586   2592   2315   2827   2837   2849   5977
                       6926   6950   6974   6999   7023   7067   7088   7109   7130   7159   7188   7217   7246
                       7275   7304   7333   7362   7384   7406   7428   7925   8188   8220   8251   8283   8316
                       8353   8390   8427   8464   8501   8542   8579   8616   8648   8680   8712   9800   9827
                       9882   9909   9936   9962   9989  10044  10073  15686  15697#
DWTB    063756         1844   1846   2200   2713   2765   2899   2907   2934   2970   7898  15687  15879#
EA017   002164          794    795    796    799#
EA020   002216          822#
EA021   002246          844#
EA023   002334          880    881    882    885#
EA032   002540          990#
EA033   002600         1012#
EA034   002640         1033#
EA041   003004         1114#
EA046   003204         1212#
EMEIS1  067021          500  16323#
EMEIS2  067052          507  16328#
EMTVEC= 000030          129#
EM1     064564          278  16106#
EM10    064650          313  16118#
EM11    065215          318  16161#
EM12    065271          324  16170#
EM13    065330          330  16176#
EM14    065362          336  16181#
EM15    065402          342  16184#
EM16    066771          349  16318#
EM17    065445          395  16190#
EM2     064564          283  16107#
EM21    065471          369  16194#
EM22    065514          375  16138#
EM23    065543          381  16202#
EM24    065562          388  16205#
EM26    065641          363    402  16213#
EM27    065666          409  16217#
EM3     064737          288  16128#
EM30    065736          416  16224#
EM31    066541          423  16292#
EM32    066050          430  16237#
EM33    066101          437  16242#
EM34    066157          444  16250#
EM35    066214          451  16255#
EM36    066302          458  16264#
EM37    066357          465  16272#
```

```
EM4      064564             293   16108#
EM40     066452             472   16282#
EM41     066001             479   16230#
EM42     066566             486   16296#
EM43     066654             493   16305#
EM45     066742             356   16314#
EM46     067100             514   16332#
EM5      064612             298   16113#
EM6      064604             303   16112#
EM7      064574             308   16110#
EOP1     065027           14992   16139#
EOP2     065045           14995   16142#
ERRA     061764            3718   15351#
ERRFLG   063164            3720*   3725*  15351*  15669#
ERRVEC=  000004             122#  14199*  14212*  15241   15242*  15244*  15247*
EX002    001640             591#
E001     001632             581#
E003     001656             601     602     603     606#
E004     001674             616     617     618     621#
E005     001712             631     632     633     636#
E006     001724             650#
E007     001740             664#
E010     001756             677     680#
E011     001776             693     696#
E012     002016             711#
E013     002042             728#
E014     002064             743#
E016     002136             776     777     778     781#
E022     002304             862     863     864     867#
E024     002370             906#
E025     002410             917     920#
E026     002430             930     933#
E027     002450             947#
E030     002470             961#
E031     002510             974#
E035     002670            1053#
E036     002710            1067#
E037     002730            1081#
E040     002756            1098#
E042     003034            1136#
E043     003056            1153#
E044     003104            1169#
E045     003132            1184#
E1015A   002102             757#
E2002    001644             595#
E2015    002112             764#
E2017    002176             806#
E2020    002226             828#
E2021    002260             851#
E2023    002344             891#
E2032    002550             996#
E2033    002610            1018#
E2034    002650            1039#
E2041    003014            1120#
E2046    003216            1218#
FAC0.0   064250           15962#
```

```
FAC0.1  064252          15963#
FAC0.2  064254          15964#
FAC0.4  064256          15965#
FAC0.5  064260          15966#
FAC1.0  064162          15933#
FAC1.1  064164          15934#
FAC1.2  064166          15935#
FAC1.3  064170          15936#
FAC1.4  064172          15937#
FAC1.5  064174          15938#
FAC2.0  064216          15949#
FAC2.1  064220          15950#
FAC2.2  064222          15951#
FAC2.3  064224          15952#
FAC2.4  064226          15953#
FAC2.5  064230          15954#
FAC3.0  064126          15919#
FAC3.1  064130          15920#
FAC3.2  064132          15921#
FAC3.3  064134          15922#
FAC3.4  064136          15923#
FAC3.5  064140          15924#
FDST0   064264          15968#
FDST1   064200          15940#
FDST2   064234          15956#
FDST3   064144          15926#
FEA     064262          15967#
FLAG    064374          16034#
FPSHI   064176          15939#
GNS   = ****** U          140   15657   15658   15659   15660
GNWHAM  064152          15929#
HI    = 000200            529#  14776   14786
HT    = 000011             32#  15448   15489   16112   16113   16118   16128   16135   16137   16142   16164   16340   16346
                        16349   16354   16360   16364   16367   16376   16381   16383   16386
IBUF    063212           3876    3897   15684#
IDENT1  065065           3933   16145#
INCRSV  061126          15127   15142#
INIT    003236           1227#  15008   15141
INSTAB  063562          13153   15816#
IOTVEC= 000020            127#
I002    001642            587     593#
I004    001664            616#
I005    001702            631#
I006    001720            646#
I007    001734            660#
I010    001750            675#
I011    001766            690#
I012    002006            706#
I013    002030            722#
I014    002060            739#
I015    002076            753#
I016    002124            774#
I017    002152            792#
I020    002212            818#
I021    002242            840#
I022    002266            860#
```

F10


```
I023    002322            878#
I024    002360            902#
I025    002376            915#
I026    002416            928#
I027    002442            943#
I030    002462            957#
I031    002502            970#
I032    002534            986#
I033    002574           1008#
I034    002634           1029#
I035    002664           1049#
I036    002704           1063#
I037    002724           1077#
I040    002746           1093#
I041    002774           1109#
I042    003026           1131#
I043    003050           1148#
I044    003074           1164#
I045    003122           1179#
I046    003166           1203#
JAM     064370          16030#
JMPS    041360          10775   10790   10796#
JMPSA   041426          10805   10822#
LCDTA   064362          16024#
LCUA    064274          15974#
LF    = 000012             33#  15483   15489   16153   16157
LFGIN   064276          15975#
LJAM    064266          15971#
LKCSR = 177546            568#  11600   11623   11641   11660   11690   11903   11934   11965   11996   12027   12062   12127
                        12174
LO    = 000100            528#  14776   14786
LPBA    064272          15973#
LSERV   064270          15972#
LTAG    064302          15977#
LWHAM   064300          15976#
MBUF0   063236           1340   1342*   1356   1358*   1417   1435   1455   1474   1491   1496   1508   1513   1531
                         1536   1554    1559   1571    1575   1597   1608   1649   1655*  1668   1673*  1703   1707*
                         1720   1722    1951   1953    1970   2044   2078   2080   2104   2105   2128   2134*  2147
                         2153*  2210*   2211   2230*   2231   2243   2258   2264   2282   2288   2300   2305*  2318
                         2323*  2393    2412   2413    2611   2636   2661   2686   2711   2726   2737   2752   2763
                         2766   2778    2789   2792    2804   2932   2938*  2950   2956*  2968   2974*  2990   2996*
                         3404   3408*   3421*  3425    3780*  3781*  4282   4302   4317   4322   4335   4352   4378
                         4448   4468    4482   4487    4500   4604   4631   4658   4685   4868   4895   4923   4951
                         4979   5007    5035   5047    5065   5095   5097   5108   5126   5128   5152   5154   5165
                         5183   5185    5209   5221    5239   5886   5914   5947   5975   5989   6007   6034   6061
                         6088   6115    6142   6169    6195   6222   6249   6276   6303   6330   6357   6384   6411
                         6438   6465    6492   6519    6676   6678   6685   6703   6705   6725   6733   6751   6772
                         6774   6781    6803   6805    6825   6833   6851   6976*  6982   7025*  7031   7048   7069*
                         7087   7108    7129   7146    7158   7175   7187   7216   7245   7248   7274   7277   7303
                         7332   7361    7364   7383    7386   7405   7427   7558   7587   7620   7649   7811   7840
                         7869   7978    8008   8039    8053   8065   8091   8093   8117   8143   8145   8164   8187
                         8219   8250    8282   8315    8331   8352   8368   8389   8426   8463   8465   8500   8502
                         8541   8578    8615   8617    8647   8649   8679   9711   8741   8762   8783   8804   8806
                         8825   8849    9010   9031    9046   9058   9085   9089   9112   9139   9143   9161   9183
                         9205   9227    9249   9252    9271   9274   9540   9569   9598   9627   9656   9685   9713
                         9742   9771   10042  10071   10100  10130  10160  10194  10224  10254  10284  10332  10353
```

```
                        10375  10397  10400  10419  10441  10444  10463  10485  10506  10509  10528  10549  10552
                        12182* 12204* 12207  13158  13237* 13243  13637  13668  13688  13705* 13740  13741* 13760
                        13777* 13812  13844* 15690  15693#
MBUF1   063242           1454   1473   2011   2016*  2229   2610   2635   2660   2685   7141*  7145   7170*  7174
                         7199*  7203   7228*  7232   7257*  7261   7286*  7290   7315*  7319   7344*  7348   7952
                         7980   8010   8847   9854   10016  10102  10132  10162  10196  10226  10256  10286  12183*
                        12213  12220* 12452  12462  12491  12501  13159  13203* 13209  15691  15695#
MD      064364          16026#
MED   = 076600            519#  13096  14203  14216  14247  14252  14255  14259  14293  14306  14341  14375  14378
                        14380  14383  14394  14407  14410  14414  14426  14429  14431  14452  14461  14476  14483
                        14495  14522  14524  14526  14531  14537  14543  14549  14555  14588  14593  14627  14643
                        14657  14674  14705  14746  14752  14761  14773  14794  14805  14819  14827  14833  14864
                        14880  14902  14917  14928  14946  14961  15041
MEDHLT  055572          14218  14219  14220  14222#
MEDT1   055602          14239#
MEDT10  056132          14335#
MEDT11  056204          14371#  14433
MEDT3   056002          14290#
MEDT4   056040          14304#
MEDO    055552          14215#
MED1    055454          14196#
MEMERR= 177744            522#  14596  14600  14785
NOPS    064330          15997#
NUA     064402          16040#
OBUF    063202           3877   3898  15679#
ONCE    063200           1224*  1229   3929   3931*  15676#
ORTAB   063420          13482  13487  15758#
PABORT= 000200            527#  14738
PA1500  064106          15894#
PA1716  064104          15893#
PBA     064372          16032#
PIRQ  = 177772             39#
PIRQVE= 000240            133#
PRIFLG  063162           1227   3700*  3705*  15406*  15668#
PRINA   062126           3698  15406#
PR0   = 000000             56#
PR1   = 000040             57#
PR2   = 000100             58#
PR3   = 000140             59#
PR4   = 000200             60#
PR5   = 000240             61#
PR6   = 000300             62#
PR7   = 000340             63#
PS    = 177776             36#    37
PSW   = 177776             37#   736   750   771   788   860*  874   898   902   915   928   981   982*
                         1003  1004* 1024  1025* 1088  1104  1294* 1309* 1323  3523* 3531  3535* 3554*
                         3559  3563* 3577* 3585  3589* 3596* 3603* 3638* 3646  3650* 3657* 3664* 3677*
                         3682  3685* 3934* 3976  3982  3996  4004  4018  4024  4038  4046  11405 11412*
                        11420  11442 11449* 11457 11506 11509 11566 11569 11666* 11697* 11712* 11723* 11729
                        11741  11744* 11756* 11762 11774 11777* 11802* 11808* 11825  11832* 11859* 11865* 11882
                        11889* 11908* 11923* 11939* 11954* 11970* 11985* 12001* 12016* 12032* 12047* 12067* 12083*
                        12099* 12115* 12135* 12145* 12181* 12202* 12238* 13977* 14006  14007  14197* 14215* 14239*
                        15065*
PSWHOL  063774          15888#
PTRP1   057636          14749  14760#
PWRUP   061000          15055  15064#
```

```
PWRVEC= 000024            128#  15019*  15020*  15029*  15035*  15050*  15051*
RCSR  = 177560            564#   3787    3873
RDBR  = 177562            565#
RDFLAG= 000144            540#  14381   14408
RDLCUR= 000103            548#  14395
RDLDAT= 000106            554#  14550   14820
RDLFGI= 000104            550#  14675   14834   14947   14962
RDLJAM= 000100            542#  14462   14628   14658   14706   14881   14918
RDLPBA= 000102            546#  14477   14544   14644   14795
RDLSER= 000101            544#  14484   14774
RDLTAG= 000107            556#  14556   14806
RDLWBA= 000105            552#
RDWBAM= 000022            538#  14376   14427
RES     064403          16041#
RESTAR  061122          15130   15139   15141#  15144   15198   15205
RESVEC= 000010            123#  14200*  14213*
REV     064377          16037#
RSBERT  061074          15133#  15201
RSERR   061046           3751   13133   14213   15126#
RSMSG   065167          15132   16157#
RSVFLG  063172           3739*   3745*  15123*  15672#
RSVTST  061040           3736   15123#
RT1A    064316          15983#
RT1B    064322          15985#
RT2A    064320          15984#
RT2B    064324          15986#
RVECT   064246          15961#
RZERO   064244          15960#
R0A     064346          16014#
R0B     064352          16016#
R1A     064112          15913#
R1B     064202          15943#
R2A     064114          15914#
R2B     064204          15944#
R3A     064116          15915#
R3B     064206          15945#
R4A     064120          15916#
R4B     064210          15946#
R5A     064122          15917#
R5B     064212          15947#
R6A     064124          15918#
R6B     064214          15948#
R7A     064350          14304   16015#
R7B     064354          14316   16017#
SCOFLG  063170           3614*   3621*  15285*  15671#
SCOPEA  061536           3615   15285#
SELTST  063166          15231*  15232*  15233   15670#
SERV    064371          16031#
SOBERR  042752          11270   11279#
SO81    042724          11265#  11348
SO82    042740          11260   11263   11273#
SO83    043116          11265   11346#
SO84    043120          11344   11348#
SO85    042732          11269#  11273
SREG    064400          16038#
STACK = 001000             27#    735    1215    1231   14198   15064
```

```
START   001630          144     578#  15066
STGBLK  064002        15891#
STKLMT= 177774           38#
SWB01 = 000071          558#  14371   14389   14393
SWR     001040          205#   3926#   3927   15027   15043#  15129   15138   15197   15229   15231   15236   15254   15261
                      15324   15336   15339
SWREG   000176          142#
SW0   = 000001           91#
SW00  = 000001           81#     91
SW01  = 000002           80#     90
SW02  = 000004           79#     89
SW03  = 000010           78#     88
SW04  = 000020           77#     87
SW05  = 000040           76#     86
SW06  = 000100           75#     85
SW07  = 000200           74#     84
SW08  = 000400           73#     83
SW09  = 001000           72#     82
SW1   = 000002           90#
SW10  = 002000           71#  15229
SW11  = 004000           70#
SW12  = 010000           69#   3927   15129   15197
SW13  = 020000           68#
SW14  = 040000           67#
SW15  = 100000           66#
SW2   = 000004           89#
SW3   = 000010           88#
SW4   = 000020           87#
SW5   = 000040           86#
SW6   = 000100           85#
SW7   = 000200           84#
SW8   = 000400           83#
SW9   = 001000           82#
TABBEG  063776        15889#
TABEND  064000        15890#
TAG   = 000040          530#  14776   14786
TBITVE= 000014          124#
TBL2    064112        14240   15910#
TBL3    064330        14290   15996#
TBL4    064346        16013#
TBL5    064362        14400   16022#
TBL6    064410        14401   16050#
TBL7    064462        14337   16078#
TBSER   061014        13141   15072#
TKVEC = 000060          131#
TLOC1   063772        14513   14528   14579   14586   14740   14750   14796   14800   14808   15887#
TLOC2   064110        15895#
TPVEC = 000064          132#
TRAPVE= 000034          130#
TRTVEC= 000014          125#
TSET    052362        13141#
TST0    003272         1239#   3935
TST1    003306         1244    1252#
TST10   003506         1347    1354#
TST100  007154         2525    2534#
TST101  007250         2562    2571#
```

TST102	007342	2599	2608#
TST103	007422	2626	2633#
TST104	007502	2651	2658#
TST105	007562	2676	2683#
TST106	007642	2701	2709#
TST107	007720	2727	2735#
TST11	003540	1363	1371#
TST110	007776	2753	2761#
TST111	010056	2779	2787#
TST112	010136	2805	2813#
TST113	010200	2828	2835#
TST114	010242	2850	2858#
TST115	010332	2894#	
TST116	010422	2930#	
TST117	010464	2941	2948#
TST12	003576	1385	1393#
TST120	010526	2959	2966#
TST121	010570	2977	2984#
TST122	010644	2999	3007#
TST123	010670	3015	3023#
TST124	010726	3043	3051#
TST125	010744	3057	3065#
TST126	010760	3071	3079#
TST127	010774	3085	3093#
TST13	003632	1407	1415#
TST130	011014	3100	3108#
TST131	011034	3115	3123#
TST132	011052	3130	3138#
TST133	011070	3144	3152#
TST134	011106	3159	3167#
TST135	011124	3174	3182#
TST136	011144	3189	3197#
TST137	011164	3204	3212#
TST14	003664	1425	1433#
TST140	011204	3219	3227#
TST141	011224	3234	3242#
TST142	011240	3248	3256#
TST143	011260	3263	3271#
TST144	011300	3278	3286#
TST145	011320	3293	3301#
TST146	011336	3308	3316#
TST147	011352	3322	3330#
TST15	003720	1443	1451#
TST150	011372	3337	3345#
TST151	011412	3352	3360#
TST152	011432	3367	3375#
TST153	011500	3393	3401#
TST154	011542	3411	3419#
TST155	011576	3428	3436#
TST156	011664	3464	3472#
TST157	011720	3485	3493#
TST16	003764	1462	1470#
TST160	011766	3509	3517#
TST161	012054	3540	3548#
TST162	012126	3561	3570#
TST163	012250	3601	3610#

TST164	012316	3631#	
TST165	012440	3662	3671#
TST166	012514	3694#	
TST167	012560	3706	3714#
TST17	004030	1481	1489#
TST170	012624	3726	3733#
TST171	012706	3757#	
TST172	012756	3770	3778#
TST173	013056	3806#	
TST174	013116	3816	3827#
TST175	013150	3836	3845#
TST176	013204	3854	3871#
TST177	013516	3943#	
TST2	003324	1258	1266#
TST20	004064	1499	1506#
TST200	013540	3950	3957#
TST201	013560	3964	3971#
TST202	013620	3984	3991#
TST203	013662	4006	4013#
TST204	013720	4026	4033#
TST205	013760	4048	4055#
TST206	014002	4062	4069#
TST207	014024	4077	4084#
TST21	004130	1521	1529#
TST210	014046	4092	4099#
TST211	014070	4107	4114#
TST212	014114	4122	4129#
TST213	014136	4136	4143#
TST214	014160	4151	4158#
TST215	014202	4166	4173#
TST216	014224	4181	4188#
TST217	014272	4207	4214#
TST22	004170	1544	1552#
TST220	014334	4228	4235#
TST221	014402	4254	4261#
TST222	014434	4271	4278#
TST223	014562	4323	4330#
TST224	014616	4340	4347#
TST225	014666	4366	4373#
TST226	014724	4384	4391#
TST227	014774	4410	4417#
TST23	004224	1561	1569#
TST230	015044	4436	4443#
TST231	015174	4488	4495#
TST232	015250	4515	4522#
TST233	015314	4541	4548#
TST234	015364	4567	4574#
TST235	015432	4593	4600#
TST236	015502	4620	4627#
TST237	015556	4647	4654#
TST24	004264	1583	1591#
TST240	015632	4674	4681#
TST241	015704	4701	4708#
TST242	015754	4727	4734#
TST243	016022	4753	4760#
TST244	016072	4779	4786#

TST245	016140	4805	4812#
TST246	016210	4831	4838#
TST247	016260	4857	4864#
TST25	004334	1609	1617#
TST250	016336	4884	4891#
TST251	016412	4911	4919#
TST252	016470	4939	4947#
TST253	016544	4967	4975#
TST254	016622	4995	5003#
TST255	016700	5023	5031#
TST256	016760	5053	5061#
TST257	017032	5079	5087#
TST26	004410	1638	1647#
TST260	017130	5114	5122#
TST261	017204	5140	5148#
TST262	017270	5171	5179#
TST263	017344	5197	5205#
TST264	017424	5227	5235#
TST265	017474	5252	5260#
TST266	017540	5279	5286#
TST267	017606	5306	5313#
TST27	004444	1658	1666#
TST270	017654	5332	5339#
TST271	017720	5357	5364#
TST272	017770	5383	5390#
TST273	020036	5409	5416#
TST274	020104	5435	5442#
TST275	020154	5461	5468#
TST276	020222	5487	5494#
TST277	020272	5513	5520#
TST3	003340	1271	1278#
TST30	004502	1676	1684#
TST300	020336	5538	5545#
TST301	020404	5564	5571#
TST302	020454	5590	5597#
TST303	020520	5616	5623#
TST304	020570	5642	5649#
TST305	020640	5667	5674#
TST306	020704	5693	5700#
TST307	020754	5719	5726#
TST31	004526	1692	1700#
TST310	021022	5745	5752#
TST311	021072	5771	5778#
TST312	021140	5797	5804#
TST313	021210	5823	5830#
TST314	021260	5849	5856#
TST315	021326	5875	5882#
TST316	021400	5902	5910#
TST317	021456	5931	5939#
TST32	004562	1710	1718#
TST320	021544	5963	5971#
TST321	021626	5995	6003#
TST322	021704	6022	6030#
TST323	021760	6049	6057#
TST324	022034	6076	6084#
TST325	022112	6103	6111#

TST326	022166	6130	6138#
TST327	022244	6157	6165#
TST33	004622	1729	1737#
TST330	022316	6183	6191#
TST331	022372	6210	6218#
TST332	022450	6237	6245#
TST333	022522	6264	6272#
TST334	022600	6291	6299#
TST335	022656	6318	6326#
TST336	022730	6345	6353#
TST337	023006	6372	6380#
TST34	004656	1750	1758#
TST340	023062	6399	6407#
TST341	023140	6426	6434#
TST342	023214	6453	6461#
TST343	023272	6480	6488#
TST344	023350	6507	6515#
TST345	023424	6534	6542#
TST346	023474	6561	6568#
TST347	023544	6587	6594#
TST35	004712	1771	1778#
TST350	023614	6613	6620#
TST351	023664	6639	6646#
TST352	023734	6665	6672#
TST353	024014	6691	6699#
TST354	024064	6713	6721#
TST355	024140	6739	6747#
TST356	024204	6760	6768#
TST357	024264	6787	6795#
TST36	004746	1791	1799#
TST360	024346	6813	6821#
TST361	024422	6839	6847#
TST362	024466	6860	6868#
TST363	024540	6888	6895#
TST364	024614	6915	6922#
TST365	024662	6939	6946#
TST366	024730	6963	6970#
TST367	025004	6988	6995#
TST37	005002	1812	1820#
TST370	025052	7012	7019#
TST371	025126	7037	7044#
TST372	025166	7056	7063#
TST373	025232	7076	7083#
TST374	025276	7096	7104#
TST375	025342	7117	7125#
TST376	025430	7147	7154#
TST377	025516	7176	7183#
TST4	003356	1284	1292#
TST40	005042	1834	1842#
TST400	025606	7205	7212#
TST401	025676	7234	7241#
TST402	025764	7263	7270#
TST403	026052	7292	7299#
TST404	026142	7321	7328#
TST405	026232	7350	7357#
TST406	026304	7371	7379#

TST407	026356	7393	7401#
TST41	005106	1857	1866#
TST410	026426	7415	7423#
TST411	026476	7437	7445#
TST412	026542	7464	7471#
TST413	026612	7491	7498#
TST414	026666	7518	7525#
TST415	026742	7545	7552#
TST416	027016	7573	7581#
TST417	027076	7602	7610#
TST42	005150	1880	1889#
TST420	027172	7635	7643#
TST421	027254	7664	7672#
TST422	027324	7692	7699#
TST423	027374	7719	7726#
TST424	027450	7746	7753#
TST425	027524	7773	7780#
TST426	027600	7800	7807#
TST427	027660	7828	7836#
TST43	005212	1903	1912#
TST430	027740	7857	7865#
TST431	030020	7885	7893#
TST432	030072	7913	7920#
TST433	030142	7940	7947#
TST434	030220	7967	7974#
TST435	030304	7996	8004#
TST436	030372	8027	8035#
TST437	030446	8054	8061#
TST44	005250	1933#	
TST440	030524	8080	8087#
TST441	030600	8106	8113#
TST442	030656	8132	8139#
TST443	030726	8152	8160#
TST444	030776	8173	8181#
TST445	031062	8204	8213#
TST446	031146	8236	8245#
TST447	031232	8268	8277#
TST45	005276	1941	1949#
TST450	031316	8300	8309#
TST451	031412	8337	8346#
TST452	031506	8374	8383#
TST453	031604	8411	8420#
TST454	031702	8448	8457#
TST455	031776	8485	8494#
TST456	032072	8522	8531#
TST457	032202	8563	8572#
TST46	005340	1960	1968#
TST460	032300	8600	8609#
TST461	032370	8632	8641#
TST462	032460	8664	8673#
TST463	032550	8696	8705#
TST464	032640	8728	8737#
TST465	032700	8750	8758#
TST466	032740	8771	8779#
TST467	033002	8792	8800#
TST47	005374	1978	1986#

TST470	033044	8813	8821#
TST471	033106	8834	8843#
TST472	033152	8856	8864#
TST473	033224	8884	8891#
TST474	033276	8912	8919#
TST475	033334	8932	8939#
TST476	033374	8951	8958#
TST477	033442	8975	8982#
TST5	003400	1299	1307#
TST50	005436	2001	2009#
TST500	033510	8999	9006#
TST501	033554	9019	9027#
TST502	033632	9047	9054#
TST503	033712	9073	9081#
TST504	033770	9100	9108#
TST505	034050	9127	9135#
TST506	034122	9149	9157#
TST507	034174	9171	9179#
TST51	005474	2018	2026#
TST510	034242	9193	9201#
TST511	034310	9215	9223#
TST512	034360	9237	9245#
TST513	034430	9259	9267#
TST514	034502	9281	9289#
TST515	034556	9309	9316#
TST516	034636	9340	9347#
TST517	034712	9367	9374#
TST52	005520	2041#	
TST520	034760	9394	9401#
TST521	035036	9421	9429#
TST522	035110	9449	9456#
TST523	035162	9476	9483#
TST524	035234	9503	9510#
TST525	035306	9529	9536#
TST526	035370	9557	9565#
TST527	035444	9586	9594#
TST53	005554	2051	2059#
TST530	035526	9615	9623#
TST531	035602	9644	9652#
TST532	035664	9673	9681#
TST533	035746	9702	9709#
TST534	036030	9730	9738#
TST535	036110	9759	9767#
TST536	036170	9787	9795#
TST537	036244	9815	9822#
TST54	005606	2068	2076#
TST540	036314	9842	9849#
TST541	036374	9870	9877#
TST542	036444	9897	9904#
TST543	036516	9924	9931#
TST544	036570	9951	9957#
TST545	036642	9977	9984#
TST546	036714	10004	10011#
TST547	036772	10031	10038#
TST55	005654	2093	2101#
TST550	037054	10059	10067#

```
TST551  037132          10088   10096#
TST552  037220          10118   10126#
TST553  037300          10148   10156#
TST554  037366          10178   10186#
TST555  037466          10212   10220#
TST556  037552          10242   10250#
TST557  037636          10272   10280#
TST56   005724           2118    2126#
TST560  037722          10301   10309#
TST561  037760          10321   10328#
TST562  040024          10341   10349#
TST563  040072          10363   10371#
TST564  040142          10385   10393#
TST565  040212          10407   10415#
TST566  040262          10429   10437#
TST567  040334          10451   10459#
TST57   005764           2137    2145#
TST570  040406          10473   10481#
TST571  040450          10494   10502#
TST572  040520          10516   10524#
TST573  040562          10537   10545#
TST574  040632          10559   10567#
TST575  040672          10577   10583   10589#
TST576  040732          10599   10604   10611#
TST577  041002          10621   10631   10638#
TST6    003420           1313    1321#
TST60   006036           2162    2169#
TST600  041042          10648   10653   10660#
TST601  041120          10670   10680   10683   10692#
TST602  041166          10702   10707   10710   10718#
TST603  041244          10728   10732   10742   10749#
TST604  041304          10759   10764   10771#
TST605  041362          10781   10791   10794   10801#
TST606  041430          10811   10816   10819   10827#
TST607  041476          10837   10842   10845   10853#
TST61   006100           2184    2192#
TST610  041544          10863   10868   10871   10879#
TST611  041620          10889   10892   10897   10900   10909#
TST612  041674          10919   10922   10927   10930   10939#
TST613  041752          10955   10967#
TST614  042040          10998#
TST615  042122          11027#
TST616  042172          11052#
TST617  042250          11068   11081#
TST62   006132           2201    2208#
TST620  042334          11097   11114#
TST621  042416          11133   11146#
TST622  042502          11162   11179#
TST623  042562          11195   11207#
TST624  042650          11224   11241#
TST625  042700          11249   11257#
TST626  042754          11275   11278   11284#
TST627  043010          11296   11303#
TST63   006172           2218    2226#
TST630  043044          11315   11322#
TST631  043100          11334   11341#
```

TST632	043134	11353	11360#			
TST633	043236	11390	11401#			
TST634	043342	11427	11438#			
TST635	043450	11464	11475#			
TST636	043626	11491	11499	11511	11520	11535#
TST637	044004	11551	11559	11571	11580	11595#
TST64	006250	2244	2252#			
TST640	044066	11619#				
TST641	044122	11629	11637#			
TST642	044160	11647	11655#			
TST643	044272	11685#				
TST644	044410	11717#				
TST645	044526	11750#				
TST646	044642	11783#				
TST647	045046	11840#				
TST65	006326	2272	2280#			
TST650	045250	11897#				
TST651	045366	11928#				
TST652	045504	11959#				
TST653	045622	11990#				
TST654	045740	12021#				
TST655	046056	12052#				
TST656	046210	12088#				
TST657	046330	12123#				
TST66	006362	2290	2298#			
TST660	046470	12168#				
TST661	046770	12251#				
TST662	047036	12271#				
TST663	047172	12311#				
TST664	047306	12349#				
TST665	047370	12374#				
TST666	047452	12399#				
TST667	047534	12424#				
TST67	006420	2308	2316#			
TST670	047620	12448#				
TST671	047726	12479	12487#			
TST672	050034	12517	12525#			
TST673	050174	12556	12577#			
TST674	050356	12625	12637#			
TST675	050540	12685	12697#			
TST676	050634	12727#				
TST677	050740	12759#				
TST7	003452	1331	1338#			
TST70	006460	2326	2334#			
TST700	051044	12791#				
TST701	051124	12814#				
TST702	051204	12837#				
TST703	051264	12859#				
TST704	051364	12892#				
TST705	051470	12910	12927#			
TST706	051576	12945	12962#			
TST707	051704	12980	12997#			
TST71	006510	2343	2351#			
TST710	051772	13023#				
TST711	052060	13050#				
TST712	052146	13077#				

```
TST713  052244          13107#
TST714  052376          13150#
TST715  052466          13177#
TST716  052516          13192#
TST717  052604          13217#
TST72   006560           2375#
TST720  052640          13233#
TST721  052702          13250#
TST722  052750          13299#
TST723  053034          13359#
TST724  053120          13419#
TST725  053204          13479#
TST726  053270          13525#
TST727  053344          13543   13569#
TST73   006610           2383    2391#
TST730  053420          13587   13599#
TST731  053504          13622   13629#
TST732  053574          13653   13660#
TST733  053672          13689   13696#
TST734  053776          13725   13732#
TST735  054102          13761   13768#
TST736  054206          13797   13804#
TST737  054300          13828   13835#
TST74   006646           2401    2409#
TST740  054362          13852   13859#
TST741  054446          13881   13888#
TST742  054534          13910   13917#
TST743  054616          13939   13946#
TST744  054670          13961   13972#
TST745  055006          14004   14014#
TST746  055056          14032   14038#
TST747  055126          14057   14063#
TST75   006710           2420    2428#
TST750  055176          14081   14087#
TST751  055264          14109   14115#
TST752  055346          14138   14144#
TST753  055446          14170   14193#
TST754  055574          14236#
TST755  055774          14287#
TST756  056124          14331#
TST757  056176          14368#
TST76   006766           2452    2460#
TST760  056500          14446#
TST761  056656          14509#
TST762  057066          14576#
TST763  057220          14617#
TST764  057426          14693#
TST765  057524          14734#
TST766  060104          14859#
TST767  060326          14934#
TST77   007060           2488    2497#
TYPE  = 104401           3703    3932   14992   14995   14998   15052   15131   15135   15199   15327   15362   15379   15381
                        15384   15386   15390   15397   15453   15550   15657#
TYPOC = 104402          14994   14997   15134   15370   15394   15658#
TYPON = 104404          15660#
TYPOS = 104403          15659#
```

F11


```
UBREAK= 177770            521#   1232#  14196#  14371#  14405#  14435#
UCNSCT  064420          16056#
UCOUNT  064446          16067#
UCUA    064432          16061#
UDCS0   064454          16070#
UDCS1   064460          16072#
UDREG   064440          16064#
UFLAG   064434          16062#
UINIT   064456          16071#
UJAM    064424          16058#
ULCDTA  064410          16052#
UM    = 140000            520#
UMD     064414          16054#
UNUA    064450          16068#
UPBA    064430          16060#
URES    064452          16069#
UREV    064442          16065#
UR6A    064142          15925#
UR6B    064232          15955#
USERV   064426          16059#
USREG   064444          16066#
VADR    064102          15892#
WCNSSW= 000226            537#   15042
WCSADR  064242          15959#
WCSA.0  064146          15927#
WCSA.1  064150          15928#
WCSB.0  064236          15957#
WCSB.1  064240          15958#
WRFLAG= 000344            541#   14384   14411   14430
WRLCUA= 000303            549#
WRLDAT= 000306            555#   14525
WRLFGI= 000304            551#
WRLJAM= 000300            543#
WRLPBA= 000302            547#   14523
WRLSER= 000301            545#
WRLTAG= 000307            557#   14527
WRLWAA= 000305            553#
WRWAAM= 000222            539#   14379   14432   14453   14496   14747   14828   14865   14903   14929
WWP   = 000100            525#   14515   14581   14583   14742   14744
XCSR  = 177564            566#    3808    3829    3847   11722   11745*  11755   11778*  11789   11846   12094   12175
XDBR  = 177566            567#
$APTHD  000700             155     161#
$ASTAT= ****** U        15602   15617
$ATYC   062670          15573   15575#
$ATY1   062644          15571#
$ATY3   062652          15438   15572#
$ATY4   062662          15332   15574#
$AUTOB  001034            202#
$BOADR  001022            197#
$BODAT  001026            199#
$CHARC  062412          15455*  15465*  15472   15481*  15486#
$CKSWR= ****** U        15663
$CMTAG  001000            185#
$CM1  = 000006            217#    218#    219#    220#    221#    222#    223#
$CM2  = 000014            217#    218#    219#    220#    221#    222#    223#
$CM3  = 000006            215#    217
```

```
$CMM  = 000005            223#    224#    225#    226#    227#    228#
$CPUOP  001146            252#
$CRLF   001115            231#  14998   15136   15327   15350   15362   15381   15386   15390   15454   15489
$DEVCT  001130            243#
$DORGN  060600          14988   15000   15006#
$ENDRD  060570            174   15002#
$ENDCT  060524          14990#
$ENULL  060604          15009#
$ENV    001140            248#  15329   15433   15580   15604
$ENVM   001141            249#   3924   15435   15440   15582
$EOP    060474          14981#
$EOPCT  060516          14987#  14991
$ERFLG  001003            188#  15214   15250   15252   15258*  15284   15317*  15350
$ERMAX  001015            194#  15252   15275*  15284
$ERROR  061544           3918   14941   15304#
$ERRPC  001016            195#  15321*  15322*  15323   15350   15368   16390   16392   16394   16396   16397   16399   16400
                        16402   16406   16418   16420   16423
$ERRTB  001150            275#  15376
$ERRTY  061772          15326   15361#
$ERTTL  001012            192#   1225*  14996   15137*  15320*  15350
$ESCAP  001112            229#  15225*  15227*  15274*  15342   15344   15350
$ETABL  001140            247#
$ETEND  001150            167     259#
$FATAL  001122            240#  15608#
$FFLG   063110          15571#  15574#  15602   15611#  15619#
$FILLC  001056            213#  15458   15489
$FILLS  001055            212#  15489
$GDADR  001020            196#
$GDDAT  001024            198#
$GET42  060560          14999#
$GTSWR= ****** U        15662
$HD   = 000000             12
$HIBTS  000700            162#
$ICNT   001004            189#  15265*  15266   15268*  15283
$ILLUP  060762          15019   15035   15057#
$INTAG  001035            203#
$ITEMB  001014            193#  15323*  15331   15350   15365
$LF     001116            232#  15350   15489
$LFLG   063107          15612#  15618#
$LPADR  001006            190#   3935*  15256*  15272*  15277   15283
$LPERR  001010            191#  10944*  10972*  11007*  11032*  11057*  11086*  11119*  11151*  11184*  11212*  11365*  11407*
                        11444*  11601*  11662*  11691*  11788*  11845*  11902*  11933*  11964*  11995*  12026*  12061*  12093*
                        12129*  12173*  12256*  12429*  12796*  12819*  12842*  12864*  13004*  13019*  13030*  13045*  13057*
                        13072*  13084*  13102*  13114*  13132*  13156*  13172*  13201*  13255*  13303*  13325*  13363*  13385*
                        13423*  13445*  13483*  13505*  13531*  13550*  13575*  13594*  14402*  14433*  15256   15273*  15283
                        15341
$MAIL   001120            163     167     238#  15271   15329   15433
$MBADR  000702            163#
$MFLG   063106          15572#  15578   15613#  15617#
$MSGAD  001134            245#  15588*  15591
$MSGLG  001136            246#  15593*
$MSGTY  001120            239#  15586   15594*  15606   15610*
$MXCNT  061534          15269   15283#
$NULL   001054            211#  15460   15489
$NWTST= 000001           1236#   1249#   1263#   1275#   1289#   1304#   1318#   1335#   1351#   1368#   1390#   1412#   1430#
                         1448#   1467#   1486#   1503#   1526#   1549#   1566#   1588#   1614#   1644#   1663#   1681#   1697#
```

```
                                1715#  1734#  1755#  1775#  1796#  1817#  1839#  1863#  1886#  1909#  1930#  1946#  1965#
                                1983#  2006#  2023#  2038#  2056#  2073#  2098#  2123#  2142#  2166#  2189#  2205#  2223#
                                2249#  2277#  2295#  2313#  2331#  2348#  2372#  2398#  2406#  2425#  2457#  2494#  2531#
                                2568#  2605#  2630#  2655#  2680#  2706#  2732#  2758#  2784#  2810#  2832#  2855#  2891#
                                2927#  2945#  2963#  2981#  3004#  3020#  3048#  3062#  3076#  3090#  3105#  3120#  3135#
                                3149#  3164#  3179#  3194#  3209#  3224#  3239#  3253#  3268#  3283#  3298#  3313#  3327#
                                3342#  3357#  3372#  3398#  3416#  3433#  3469#  3490#  3514#  3545#  3567#  3607#  3628#
                                3668#  3691#  3711#  3730#  3754#  3775#  3803#  3824#  3842#  3860#  3862   3940#  3954#
                                3968#  3988#  4010#  4030#  4052#  4066#  4081#  4096#  4111#  4126#  4140#  4155#  4170#
                                4185#  4211#  4232#  4258#  4275#  4327#  4344#  4370#  4388#  4414#  4440#  4492#  4519#
                                4545#  4571#  4597#  4624#  4651#  4678#  4705#  4731#  4757#  4783#  4809#  4835#  4861#
                                4888#  4916#  4944#  4972#  5000#  5028#  5058#  5084#  5119#  5145#  5176#  5202#  5232#
                                5257#  5283#  5310#  5336#  5361#  5387#  5413#  5439#  5465#  5491#  5517#  5542#  5568#
                                5594#  5620#  5646#  5671#  5697#  5723#  5749#  5775#  5801#  5827#  5853#  5879#  5907#
                                5936#  5968#  6000#  6027#  6054#  6081#  6108#  6135#  6162#  6188#  6215#  6242#  6269#
                                6296#  6323#  6350#  6377#  6404#  6431#  6458#  6485#  6512#  6539#  6565#  6591#  6617#
                                6643#  6669#  6696#  6718#  6744#  6765#  6792#  6818#  6844#  6865#  6892#  6919#  6943#
                                6967#  6992#  7016#  7041#  7060#  7080#  7101#  7122#  7151#  7180#  7209#  7238#  7267#
                                7296#  7325#  7354#  7376#  7398#  7420#  7442#  7468#  7495#  7522#  7549#  7578#  7607#
                                7640#  7669#  7696#  7723#  7750#  7777#  7804#  7833#  7862#  7890#  7917#  7944#  7971#
                                8001#  8032#  8058#  8084#  8110#  8136#  8157#  8178#  8210#  8242#  8274#  8306#  8343#
                                8380#  8417#  8454#  8491#  8528#  8569#  8606#  8638#  8670#  8702#  8734#  8755#  8776#
                                8797#  8818#  8840#  8861#  8888#  8916#  8936#  8955#  8979#  9003#  9024#  9051#  9078#
                                9105#  9132#  9154#  9176#  9198#  9220#  9242#  9264#  9286#  9313#  9344#  9371#  9398#
                                9426#  9453#  9480#  9507#  9533#  9562#  9591#  9620#  9649#  9678#  9706#  9735#  9764#
                                9792#  9819#  9846#  9874#  9901#  9928#  9954#  9981#  10008#  10035#  10064#  10093#  10123#
                               10153#  10183#  10217#  10247#  10277#  10306#  10325#  10346#  10368#  10390#  10412#  10434#  10456#
                               10478#  10499#  10521#  10542#  10564#  10586#  10608#  10635#  10657#  10689#  10715#  10746#  10768#
                               10798#  10824#  10850#  10876#  10906#  10936#  10964#  10995#  11024#  11049#  11078#  11111#  11143#
                               11176#  11204#  11238#  11254#  11281#  11300#  11319#  11338#  11357#  11398#  11435#  11472#  11532#
                               11592#  11616#  11634#  11652#  11682#  11714#  11747#  11780#  11837#  11894#  11925#  11956#  11987#
                               12018#  12049#  12085#  12117#  12119   12161#  12163   12248#  12268#  12308#  12346#  12371#  12396#
                               12421#  12445#  12484#  12522#  12574#  12634#  12694#  12724#  12756#  12788#  12811#  12834#  12856#
                               12887#  12889   12924#  12959#  12994#  13020#  13047#  13074#  13104#  13144#  13146   13174#  13189#
                               13214#  13230#  13247#  13267#  13269   13327#  13329   13387#  13389   13447#  13449   13507#  13509
                               13552#  13554   13596#  13626#  13657#  13693#  13729#  13765#  13801#  13832#  13856#  13885#  13914#
                               13943#  13965#  13967   14011#  14035#  14060#  14084#  14112#  14141#  14186#  14188   14224#  14226
                               14275#  14277   14321#  14323   14351#  14353   14438#  14440   14499#  14501   14569#  14571   14606#
                               14608   14683#  14685   14722#  14724   14844#  14846   14931#
$OCNT   062640                 15522*  15551*  15564#
$OMODE  062642                 15517*  15521*  15526   15529*  15540*  15566#
$OVER   061504                 15234   15237   15257   15267   15276#
$PASS   001126                   242#   1226*  14984*  14985*  14993   15009   15263   15284
$PASTM  000706                   165#
$POWER  060770                 15053   15060#
$PWRAD  060756                 15055#
$PWRON  060610                  3922   15019#  15050
$PWRNG  060752                 15053#
$PWRUP  060662                 15029   15035#
$QUES   001114                   230#  15350   15489
$RDCHR= ****** U               15663
$RDDEC= ****** U               15663
$RDLIN= ****** U               15663
$RDOCT= ****** U               15663
$REGAD  001060                   215#  15306   16404   16418   16420
$REGO   001062                   217#  14739*  14750*  14767   14837*  16392   16397   16399   16400   16406   16418   16420   16423
```

DQKDAA.P11 08-FEB-77 16:17 CROSS REFERENCE TABLE -- USER SYMBOLS

```
$REG1   001064          218#  16397   16400   16402   16404   16418   16420
$REG2   001066          219#  16400   16402   16404   16417
$REG3   001070          220#  16400   16402   16404   16416   16420
$REG4   001072          221#  16404   16415   16420
$REG5   001074          222#   1233*  11487*  11495*  11507*  11525*  11547*  11555*  11567*  11585*  12286*  12294*  12301*
                      12325*  12333*  12340*  12364*  12389*  12414*  12530*  12600*  12610*  12620*  12650*  12660*  12670*
                      12680*  12708*  12714*  12718*  12740*  12749*  12772*  12781*  15346*  16418
$RTNAD  060602        15008#
$R2A  = ****** U      15663
$SAVRE= ****** U      15663
$SAVR6  060766        15028*  15036   15037*  15038*  15059#
$SCOPE  061204         3916   14956   15222#
$SETUP= 000000        14983   15223   15305   15339   15346
$SVLAD  061450        15245   15270#
$SVPC = 000714          172#    177
$SWR  = 165000            1#     12      16      17      18      19      20      21      22      23     228     229     230
                       1241    1254    1268    1280    1294    1309    1323    1340    1356    1373    1395    1417    1435
                       1453    1472    1491    1508    1531    1554    1571    1593    1619    1649    1668    1686    1702
                       1720    1739    1760    1780    1801    1822    1844    1868    1891    1914    1935    1951    1970
                       1988    2011    2028    2043    2061    2078    2103    2128    2147    2171    2194    2210    2228
                       2254    2282    2300    2318    2336    2353    2377    2393    2411    2430    2462    2499    2536
                       2573    2610    2635    2660    2685    2711    2737    2763    2789    2815    2837    2860    2896
                       2932    2950    2968    2986    3009    3025    3053    3067    3081    3095    3110    3125    3140
                       3154    3169    3184    3199    3214    3229    3244    3258    3273    3288    3303    3318    3332
                       3347    3362    3377    3403    3421    3438    3474    3496    3519    3550    3572    3612    3633
                       3673    3696    3716    3735    3759    3780    3808    3829    3847    3873    3947    3961    3975
                       3995    4017    4037    4059    4073    4088    4103    4118    4133    4147    4162    4177    4192
                       4218    4239    4265    4282    4334    4351    4377    4395    4421    4447    4499    4526    4552
                       4578    4604    4631    4658    4685    4712    4738    4764    4790    4816    4842    4868    4895
                       4923    4951    4979    5007    5035    5065    5091    5126    5152    5183    5209    5239    5264
                       5290    5317    5343    5368    5394    5420    5446    5472    5498    5524    5549    5575    5601
                       5627    5653    5678    5704    5730    5756    5782    5808    5834    5860    5886    5914    5943
                       5975    6007    6034    6061    6088    6115    6142    6169    6195    6222    6249    6276    6303
                       6330    6357    6384    6411    6438    6465    6492    6519    6546    6572    6598    6624    6650
                       6676    6703    6725    6751    6772    6799    6825    6851    6872    6899    6926    6950    6974
                       6999    7023    7048    7067    7087    7108    7129    7158    7187    7216    7245    7274    7303
                       7332    7361    7383    7405    7427    7449    7475    7502    7529    7556    7585    7614    7647
                       7676    7703    7730    7757    7784    7811    7840    7869    7897    7924    7951    7978    8008
                       8039    8065    8091    8117    8143    8164    8185    8217    8249    8281    8313    8350    8387
                       8424    8461    8498    8535    8576    8613    8645    8677    8709    8741    8762    8783    8804
                       8825    8847    8868    8895    8923    8943    8962    8986    9010    9031    9058    9085    9112
                       9139    9161    9183    9205    9227    9249    9271    9293    9320    9351    9378    9405    9433
                       9460    9487    9514    9540    9569    9598    9627    9656    9685    9713    9742    9771    9799
                       9826    9853    9881    9908    9935    9961    9988   10015   10042   10071   10100   10130   10160
                      10190   10224   10254   10284   10313   10332   10353   10375   10397   10419   10441   10463   10485
                      10506   10528   10549   10571   10593   10615   10642   10664   10696   10722   10753   10775   10805
                      10831   10857   10883   10913   10943   10971   11002   11031   11056   11085   11118   11150   11183
                      11211   11245   11260   11288   11307   11326   11344   11364   11405   11442   11479   11539   11599
                      11623   11641   11659   11689   11721   11754   11787   11844   11901   11932   11963   11994   12025
                      12056   12092   12127   12172   12255   12275   12315   12353   12378   12403   12428   12452   12491
                      12529   12581   12641   12701   12731   12763   12795   12818   12841   12863   12896   12931   12966
                      13000   13026   13053   13080   13110   13152   13181   13196   13221   13237   13254   13302   13362
                      13422   13482   13528   13572   13602   13632   13663   13699   13735   13771   13807   13838   13862
                      13891   13920   13949   13976   14018   14042   14067   14090   14118   14147   14196   14239   14290
                      14334   14371   14449   14512   14579   14620   14696   14737   14862   14937   14978   14983   15001
                      15007   15009   15056   15215   15216   15217   15218   15219   15236   15248   15250   15251   15252
```

```
                    15259   15260   15261   15273   15276   15283   15297   15298   15299   15300   15320   15324   15336
                    15339   15350
$SWREG  001142        250#    3926
$SWRMK= 000000      15219
$TESTN  001124        241#    3937#  15223   15271#  15281#  16423
$TIMES  001110        228#    3936#  11721#  11754#  14983#  15259#  15266   15269#  15283
$TKB    001046        208#
$TKS    001044        207#
$TMP0   001076        223#   13602#  13632#  13663#  13699#  13735#  13771#  13807#  13838#  13862#  13891#  13920#  13949#
                    14090#  14118#  14151#  14249#  14258   14388#  14418#  16390   16406
$TMP1   001100        224#   14250#  14263#  14269   14310#  14345#  14389#  14419#  16390   16392   16394
$TMP2   001102        225#   14241#  14251   14261   14268#  14270   14311#  14346#  16392   16394
$TMP3   001104        226#   14257#  14261   14312#  16394
$TMP4   001106        227#   14246#  14264   14267#
$TN     = 000770        1#      12    1236    1240    1241#   1244    1249    1253    1254#   1258    1263    1267    1268#
                    1271    1275    1279    1280#   1284    1289    1293    1294#   1299    1304    1308    1309#   1313
                    1318    1322    1323#   1331    1335    1339    1340#   1347    1351    1355    1356#   1363    1368
                    1372    1373#   1385    1390    1394    1395#   1407    1412    1416    1417#   1425    1430    1434
                    1435#   1443    1448    1452    1453#   1462    1467    1471    1472#   1481    1486    1490    1491#
                    1499    1503    1507    1508#   1521    1526    1530    1531#   1544    1549    1553    1554#   1561
                    1566    1570    1571#   1583    1588    1592    1593#   1609    1614    1618    1619#   1638    1644
                    1648    1649#   1658    1663    1667    1668#   1676    1681    1685    1686#   1692    1697    1701
                    1702#   1710    1715    1719    1720#   1729    1734    1738    1739#   1750    1755    1759    1760#
                    1771    1775    1779    1780#   1791    1796    1800    1801#   1812    1817    1821    1822#   1834
                    1839    1843    1844#   1857    1863    1867    1868#   1880    1886    1890    1891#   1903    1909
                    1913    1914#   1930    1934    1935#   1941    1946    1950    1951#   1960    1965    1969    1970#
                    1978    1983    1987    1988#   2001    2006    2010    2011#   2018    2023    2027    2028#   2038
                    2042    2043#   2051    2056    2060    2061#   2068    2073    2077    2078#   2093    2098    2102
                    2103#   2118    2123    2127    2128#   2137    2142    2146    2147#   2162    2166    2170    2171#
                    2184    2189    2193    2194#   2201    2205    2209    2210#   2218    2223    2227    2228#   2244
                    2249    2253    2254#   2272    2277    2281    2282#   2290    2295    2299    2300#   2308    2313
                    2317    2318#   2326    2331    2335    2336#   2343    2348    2352    2353#   2372    2376    2377#
                    2383    2388    2392    2393#   2401    2406    2410    2411#   2420    2425    2429    2430#   2452
                    2457    2461    2462#   2488    2494    2498    2499#   2525    2531    2535    2536#   2562    2568
                    2572    2573#   2599    2605    2609    2610#   2626    2630    2634    2635#   2651    2655    2659
                    2660#   2676    2680    2684    2685#   2701    2706    2710    2711#   2727    2732    2736    2737#
                    2753    2758    2762    2763#   2779    2784    2788    2789#   2805    2810    2814    2815#   2828
                    2832    2836    2837#   2850    2855    2859    2860#   2891    2895    2896#   2927    2931    2932#
                    2941    2945    2949    2950#   2959    2963    2967    2968#   2977    2981    2985    2986#   2999
                    3004    3008    3009#   3015    3020    3024    3025#   3043    3048    3052    3053#   3057    3062
                    3066    3067#   3071    3076    3080    3081#   3085    3090    3094    3095#   3100    3105    3109
                    3110#   3115    3120    3124    3125#   3130    3135    3139    3140#   3144    3149    3153    3154#
                    3159    3164    3168    3169#   3174    3179    3183    3184#   3189    3194    3198    3199#   3204
                    3209    3213    3214#   3219    3224    3228    3229#   3234    3239    3243    3244#   3248    3253
                    3257    3258#   3263    3268    3272    3273#   3278    3283    3287    3288#   3293    3298    3302
                    3303#   3308    3313    3317    3318#   3322    3327    3331    3332#   3337    3342    3346    3347#
                    3352    3357    3361    3362#   3367    3372    3376    3377#   3393    3398    3402    3403#   3411
                    3416    3420    3421#   3428    3433    3437    3438#   3464    3469    3473    3474#   3485    3490
                    3494    3495#   3509    3514    3518    3519#   3540    3545    3549    3550#   3561    3567    3571
                    3572#   3601    3607    3611    3612#   3628    3632    3633#   3662    3668    3672    3673#   3691
                    3695    3696#   3706    3711    3715    3716#   3726    3730    3734    3735#   3754    3758    3759#
                    3770    3775    3779    3780#   3803    3807    3808#   3816    3824    3828    3829#   3836    3842
                    3846    3847#   3854    3860    3872    3873#   3940    3945    3947#   3950    3954    3959    3961#
                    3964    3968    3973    3975#   3984    3988    3993    3995#   4006    4010    4015    4017#   4026
                    4030    4035    4037#   4048    4052    4057    4059#   4062    4066    4071    4073#   4077    4081
                    4086    4088#   4092    4096    4101    4103#   4107    4111    4116    4118#   4122    4126    4131
```

```
4133#  4136   4140   4145   4147#  4151   4155   4160   4162#  4166   4170   4175   4177#
4181   4185   4190   4192#  4207   4211   4216   4218#  4228   4232   4237   4239#  4254
4258   4263   4265#  4271   4275   4280   4282#  4323   4327   4332   4334#  4340   4344
4349   4351#  4366   4370   4375   4377#  4384   4388   4393   4395#  4410   4414   4419
4421#  4436   4440   4445   4447#  4488   4492   4497   4499#  4515   4519   4524   4526#
4541   4545   4550   4552#  4567   4571   4576   4578#  4593   4597   4602   4604#  4620
4624   4629   4631#  4647   4651   4656   4658#  4674   4678   4683   4685#  4701   4705
4710   4712#  4727   4731   4736   4738#  4753   4757   4762   4764#  4779   4783   4788
4790#  4805   4809   4814   4816#  4831   4835   4840   4842#  4857   4861   4866   4868#
4884   4888   4893   4895#  4911   4916   4921   4923#  4939   4944   4949   4951#  4967
4972   4977   4979#  4995   5000   5005   5007#  5023   5028   5033   5035#  5053   5058
5063   5065#  5079   5084   5089   5091#  5114   5119   5124   5126#  5140   5145   5150
5152#  5171   5176   5181   5183#  5197   5202   5207   5209#  5227   5232   5237   5239#
5252   5257   5262   5264#  5279   5283   5288   5290#  5306   5310   5315   5317#  5332
5336   5341   5343#  5357   5361   5366   5368#  5383   5387   5392   5394#  5409   5413
5418   5420#  5435   5439   5444   5446#  5461   5465   5470   5472#  5487   5491   5496
5498#  5513   5517   5522   5524#  5538   5542   5547   5549#  5564   5568   5573   5575#
5590   5594   5599   5601#  5616   5620   5625   5627#  5642   5646   5651   5653#  5667
5671   5676   5678#  5693   5697   5702   5704#  5719   5723   5728   5730#  5745   5749
5754   5756#  5771   5775   5780   5782#  5797   5801   5806   5808#  5823   5827   5832
5834#  5849   5853   5858   5860#  5875   5879   5884   5886#  5902   5907   5912   5914#
5931   5936   5941   5943#  5963   5968   5973   5975#  5995   6000   6005   6007#  6022
6027   6032   6034#  6049   6054   6059   6061#  6076   6081   6086   6088#  6103   6108
6113   6115#  6130   6135   6140   6142#  6157   6162   6167   6169#  6183   6188   6193
6195#  6210   6215   6220   6222#  6237   6242   6247   6249#  6264   6269   6274   6276#
6291   6296   6301   6303#  6318   6323   6328   6330#  6345   6350   6355   6357#  6372
6377   6382   6384#  6399   6404   6409   6411#  6426   6431   6436   6438#  6453   6458
6463   6465#  6480   6485   6490   6492#  6507   6512   6517   6519#  6534   6539   6544
6546#  6561   6565   6570   6572#  6587   6591   6596   6598#  6613   6617   6622   6624#
6639   6643   6648   6650#  6665   6669   6674   6676#  6691   6696   6701   6703#  6713
6718   6723   6725#  6739   6744   6749   6751#  6760   6765   6770   6772#  6787   6792
6797   6799#  6813   6818   6823   6825#  6839   6844   6849   6851#  6860   6865   6870
6872#  6888   6892   6897   6899#  6915   6919   6924   6926#  6939   6943   6948   6950#
6963   6967   6972   6974#  6988   6992   6997   6999#  7012   7016   7021   7023#  7037
7041   7046   7048#  7056   7060   7065   7067#  7076   7080   7085   7087#  7096   7101
7106   7108#  7117   7122   7127   7129#  7147   7151   7156   7158#  7176   7180   7185
7187#  7205   7209   7214   7216#  7234   7238   7243   7245#  7263   7267   7272   7274#
7292   7296   7301   7303#  7321   7325   7330   7332#  7350   7354   7359   7361#  7371
7376   7381   7383#  7393   7398   7403   7405#  7415   7420   7425   7427#  7437   7442
7447   7449#  7464   7468   7473   7475#  7491   7495   7500   7502#  7518   7522   7527
7529#  7545   7549   7554   7556#  7573   7578   7583   7585#  7602   7607   7612   7614#
7635   7640   7645   7647#  7664   7668   7674   7676#  7692   7696   7701   7703#  7719
7723   7728   7730#  7746   7750   7755   7757#  7773   7777   7782   7784#  7800   7804
7809   7811#  7828   7833   7838   7840#  7857   7862   7867   7869#  7885   7890   7895
7897#  7913   7917   7922   7924#  7940   7944   7949   7951#  7967   7971   7976   7978#
7996   8001   8006   8008#  8027   8032   8037   8039#  8054   8058   8063   8065#  8080
8084   8089   8091#  8106   8110   8115   8117#  8132   8136   8141   8143#  8152   8157
8162   8164#  8173   8178   8183   8185#  8204   8210   8215   8217#  8236   8242   8247
8249#  8268   8274   8279   8281#  8300   8306   8311   8313#  8337   8343   8348   8350#
8374   8380   8385   8387#  8411   8417   8422   8424#  8448   8454   8459   8461#  8485
8491   8496   8498#  8522   8528   8533   8535#  8563   8569   8574   8576#  8600   8606
8611   8613#  8632   8638   8643   8645#  8664   8670   8675   8677#  8696   8702   8707
8709#  8728   8734   8739   8741#  8750   8755   8760   8762#  8771   8776   8781   8783#
8792   8797   8802   8804#  8813   8818   8823   8825#  8834   8840   8845   8847#  8856
8861   8866   8868#  8884   8888   8893   8895#  8912   8916   8921   8923#  8932   8936
8941   8943#  8951   8955   8960   8962#  8975   8979   8984   8986#  8999   9003   9008
```

```
                9010#   9019    9024    9029    9031#   9047    9051    9056    9058#   9073    9078    9083    9085#
                9100    9105    9110    9112#   9127    9132    9137    9139#   9149    9154    9159    9161#   9171
                9176    9181    9183#   9193    9198    9203    9205#   9215    9220    9225    9227#   9237    9242
                9247    9249#   9259    9264    9269    9271#   9281    9286    9291    9293#   9309    9313    9318
                9320#   9340    9344    9349    9351#   9367    9371    9376    9378#   9394    9398    9403    9405#
                9421    9426    9431    9433#   9449    9453    9458    9460#   9476    9480    9485    9487#   9503
                9507    9512    9514#   9529    9533    9538    9540#   9557    9562    9567    9569#   9586    9591
                9596    9598#   9615    9620    9625    9627#   9644    9649    9654    9676#   9673    9678    9683
                9685#   9702    9706    9711    9713#   9730    9735    9740    9742#   9759    9764    9769    9771#
                9787    9792    9797    9799#   9815    9819    9824    9826#   9842    9846    9851    9853#   9870
                9874    9879    9881#   9897    9901    9906    9908#   9924    9928    9933    9935#   9951    9954
                9959    9961#   9977    9981    9986    9988#   10004   10008   10013   10015#  10031   10035   10040
               10042#  10059   10064   10069   10071#  10088   10093   10098   10100#  10118   10123   10128   10130#
               10148   10153   10158   10160#  10178   10183   10188   10190#  10212   10217   10222   10224#  10242
               10247   10252   10254#  10272   10277   10282   10284#  10301   10306   10311   10313#  10321   10325
               10330   10332#  10341   10346   10351   10353#  10363   10368   10373   10375#  10385   10390   10395
               10397#  10407   10412   10417   10419#  10429   10434   10439   10441#  10451   10456   10461   10463#
               10473   10478   10483   10485#  10494   10499   10504   10506#  10516   10521   10526   10528#  10537
               10542   10547   10549#  10559   10564   10569   10571#  10577   10582   10586   10591   10593#  10599
               10604   10608   10613   10615#  10621   10631   10635   10640   10642#  10648   10653   10657   10662
               10664#  10670   10680   10683   10689   10694   10696#  10702   10707   10710   10715   10720   10722#
               10728   10732   10742   10746   10751   10753#  10759   10764   10768   10773   10775#  10781   10791
               10794   10798   10803   10805#  10811   10816   10819   10824   10829   10831#  10837   10842   10845
               10850   10855   10857#  10863   10868   10871   10876   10881   10883#  10889   10892   10897   10900
               10906   10911   10913#  10919   10922   10927   10930   10936   10941   10943#  10955   10964   10969
               10971#  10997   11000   11002#  11024   11029   11031#  11049   11054   11056#  11068   11078   11083
               11085#  11097   11111   11116   11118#  11133   11143   11148   11150#  11162   11176   11181   11183#
               11195   11204   11209   11211#  11224   11238   11243   11245#  11249   11254   11259   11260#  11275
               11278   11281   11286   11288#  11296   11300   11305   11307#  11315   11319   11324   11326#  11334
               11338   11343   11344#  11353   11357   11362   11364#  11390   11398   11403   11405#  11427   11435
               11440   11442#  11464   11472   11477   11479#  11491   11499   11511   11520   11532   11537   11539#
               11551   11559   11571   11580   11592   11597   11599#  11616   11621   11623#  11629   11634   11639
               11641#  11647   11652   11657   11659#  11682   11687   11689#  11714   11719   11721#  11747   11752
               11754#  11780   11785   11787#  11837   11842   11844#  11894   11899   11901#  11925   11930   11932#
               11956   11961   11963#  11987   11992   11994#  12018   12023   12025#  12049   12054   12056#  12085
               12090   12092#  12117   12125   12127#  12161   12170   12172#  12248   12253   12255#  12268   12273
               12275#  12308   12313   12315#  12346   12351   12353#  12371   12376   12378#  12396   12401   12403#
               12421   12426   12428#  12445   12450   12452#  12479   12484   12489   12491#  12517   12522   12527
               12529#  12556   12574   12579   12581#  12625   12634   12639   12641#  12685   12694   12699   12701#
               12724   12729   12731#  12756   12761   12763#  12788   12793   12795#  12811   12816   12818#  12834
               12839   12841#  12856   12861   12863#  12887   12894   12896#  12910   12924   12929   12931#  12945
               12959   12964   12966#  12980   12994   12999   13000#  13020   13025   13026#  13047   13052   13053#
               13074   13079   13080#  13104   13109   13110#  13144   13151   13152#  13174   13179   13181#  13189
               13194   13196#  13214   13219   13221#  13230   13235   13237#  13247   13252   13254#  13267   13301
               13302#  13327   13361   13362#  13387   13421   13422#  13447   13481   13482#  13507   13527   13528#
               13543   13552   13571   13572#  13587   13596   13601   13602#  13622   13626   13631   13632#  13653
               13657   13662   13663#  13689   13693   13698   13699#  13725   13729   13734   13735#  13761   13765
               13770   13771#  13797   13801   13806   13807#  13828   13832   13837   13838#  13852   13856   13861
               13862#  13881   13885   13890   13891#  13910   13914   13919   13920#  13939   13943   13948   13949#
               13961   13965   13974   13976#  14004   14011   14016   14018#  14032   14035   14040   14042#  14057
               14060   14065   14067#  14081   14084   14089   14090#  14109   14112   14117   14118#  14138   14141
               14146   14147#  14170   14186   14194   14196#  14224   14237   14239#  14275   14288   14290#  14321
               14332   14334#  14351   14369   14371#  14438   14447   14449#  14499   14510   14512#  14569   14577
               14579#  14606   14618   14620#  14683   14694   14696#  14722   14735   14737#  14844   14860   14862#
               14931   14935   14937#  14968
$TPB    001052          210#  15478#  15489
```

```
$TPFLG  001057            214#  15427   15489
$TPS    001050            209#  15476   15489
$TRAP   063112           3920   15634#
$TRAP2  063134          15645#  15656
$TRP  = 000005          15649#  15658#  15659#  15660#  15661#
$TRPAD  063146          15639   15656#
$TSTM   000704            164#
$TSTNM  001002            187#  15214   15228#  15270#  15271   15276   15278   15284   15319   15350
$TYPBN= ****** U        15661
$TYPDS= ****** U        15661
$TYPE   062134          15427#  15599   15649   15657
$TYPEC  062346          15457   15464   15471   15476#  15477
$TYPEX  062414          15482   15484   15487#
$TYPOC  062442          15520#  15658
$TYPON  062456          15519   15522#  15660
$TYPOS  062416          15515#  15659
$UNIT   001132            244#
$UNITM  000710            166#
$USWR   001144            251#
$XTSTR  061274          15239#
$$GET4= 000000          15001#
$DFILL  062641          15516#  15520#  15530   15565#
$40CAT= ****** U        15236   15326
.     = 067770            136#    140#    145#    151     152#    154#    156#    172     173#    175#    177#    184#    233
                         1595    2036    2256    2988    3497    4220    5093    5945    6801    7616    8537    9322   10192
                        11004   11703   11914   11945   11976   12007   12038   12058   12073   12105   13198   14149   14221
                        15009   15010#  15031   15058   15283   15284   15350   15400#  15489   15620#  15684#  15868   15869
                        15891#  16411#
.$ASTA= ****** U        15572   15575
.$X   = 000700            151#    156
```

```
COMMEN    134#
ENDCOM    134#
ENDERR  15290#  15346
ENDPAS  14973#  14992
ENDSCO  15208#  15278
ERROR      28#   3723   3952   3966   3986   4002   4008   4028   4044   4050   4064   4079   4094   4109   4124
          4138   4153   4168   4183   4204   4209   4230   4250   4256   4273   4295   4300   4315   4320   4325
          4342   4363   4368   4386   4407   4412   4433   4438   4460   4465   4480   4485   4490   4512   4517
          4538   4543   4564   4569   4590   4595   4617   4622   4644   4649   4671   4676   4698   4703   4724
          4729   4750   4755   4776   4781   4802   4807   4828   4833   4854   4859   4881   4886   4908   4914
          4936   4942   4964   4970   4992   4998   5020   5026   5045   5050   5056   5076   5082   5106   5111
          5117   5137   5143   5163   5168   5174   5194   5200   5219   5224   5230   5249   5255   5276   5281
          5303   5308   5329   5334   5354   5359   5380   5385   5406   5411   5432   5437   5458   5463   5484
          5489   5510   5515   5535   5540   5561   5566   5587   5592   5613   5618   5639   5644   5665   5669
          5690   5695   5716   5721   5742   5747   5768   5773   5794   5799   5820   5825   5846   5851   5872
          5877   5899   5905   5928   5934   5960   5966   5987   5992   5998   6020   6025   6047   6052   6074
          6079   6101   6106   6128   6133   6155   6160   6181   6186   6208   6213   6235   6240   6262   6267
          6289   6294   6316   6321   6343   6348   6370   6375   6397   6402   6424   6429   6451   6456   6478
          6483   6505   6510   6532   6537   6558   6563   6584   6589   6610   6615   6636   6641   6662   6667
          6688   6694   6716   6736   6742   6763   6784   6790   6816   6836   6842   6863   6885   6890   6912
          6917   6936   6941   6960   6965   6985   6990   7009   7014   7034   7039   7058   7078   7099   7120
          7143   7149   7172   7178   7201   7207   7230   7236   7259   7265   7288   7294   7317   7323   7346
          7352   7374   7396   7418   7440   7461   7466   7488   7493   7515   7520   7542   7547   7570   7576
          7599   7605   7632   7638   7661   7667   7689   7694   7716   7721   7743   7748   7770   7775   7797
          7802   7825   7831   7854   7860   7882   7888   7910   7915   7937   7942   7964   7969   7993   7999
          8024   8030   8051   8056   8077   8082   8103   8108   8129   8134   8155   8176   8201   8208   8233
          8240   8265   8272   8297   8304   8329   8334   8341   8366   8371   8378   8403   8408   8415   8440
          8445   8452   8477   8482   8489   8514   8519   8526   8555   8560   8567   8592   8597   8604   8629
          8636   8661   8668   8693   8700   8725   8732   8753   8774   8795   8816   8837   8859   8881   8886
          8909   8914   8934   8953   8972   8977   8996   9001   9022   9044   9049   9070   9076   9097   9103
          9124   9130   9152   9174   9196   9218   9240   9262   9284   9306   9311   9337   9342   9364   9369
          9391   9396   9418   9424   9446   9451   9473   9478   9500   9505   9526   9531   9554   9560   9583
          9589   9612   9618   9641   9647   9670   9676   9699   9705   9727   9733   9756   9762   9784   9790
          9812   9817   9839   9844   9867   9872   9894   9899   9921   9926   9948   9953   9974   9979  10001
         10006  10028  10033  10056  10062  10085  10091  10115  10121  10145  10151  10175  10181  10209  10215
         10239  10245  10269  10275  10298  10304  10323  10344  10366  10388  10410  10432  10454  10476  10497
         10519  10540  10562  10576  10584  10598  10606  10620  10628  10633  10647  10655  10669  10677  10682
         10685  10701  10709  10713  10727  10731  10739  10744  10758  10766  10780  10788  10793  10796  10810
         10818  10822  10836  10844  10847  10862  10870  10873  10888  10891  10899  10904  10918  10921  10929
         10934  10951  10961  10981  10986  10991  11014  11021  11039  11045  11064  11074  11093  11103  11107
         11126  11129  11139  11158  11168  11172  11191  11201  11219  11230  11234  11248  11277  11279  11298
         11317  11336  11355  11374  11382  11387  11394  11417  11424  11431  11454  11461  11468  11489  11497
         11510  11518  11528  11549  11557  11570  11578  11588  11611  11632  11650  11675  11706  11734  11742
         11767  11775  11810  11817  11826  11829  11867  11874  11883  11886  11917  11948  11979  12010  12041
         12077  12109  12151  12155  12210  12217  12224  12230  12235  12263  12288  12296  12303  12327  12335
         12342  12366  12391  12416  12439  12460  12468  12476  12499  12507  12514  12537  12545  12553  12570
         12592  12602  12612  12622  12652  12662  12672  12682  12710  12720  12742  12751  12774  12783  12806
         12829  12852  12874  12881  12909  12919  12944  12954  12979  12989  13012  13038  13065  13093  13122
         13169  13187  13211  13228  13245  13265  13320  13380  13440  13500  13541  13585  13617  13624  13648
         13655  13679  13686  13691  13715  13722  13727  13751  13758  13763  13787  13794  13799  13823  13830
         13854  13876  13883  13905  13912  13934  13941  13963  14009  14029  14033  14054  14058  14078  14082
         14104  14110  14133  14139  14165  14171  14205  14209  14222  14266  14313  14347  14390  14399  14420
         14457  14467  14472  14491  14534  14563  14590  14601  14633  14640  14648  14662  14670  14678  14700
         14711  14717  14754  14770  14779  14790  14801  14816  14824  14638  14886  14897  14920  14950  14965
         15226
ESCAPE    134#
```

```
GETPRI    134#
GETSWR    134#
MSG      3860#  3862  12117#  12119  12161#  12163  12887#  12889  13144#  13146  13267#  13269  13327#  13329  13387#
        13389  13447#  13449  13507#  13509  13552#  13554  13965#  13967  14437#  14440  14498#  14501  14568#  14571
        14605#  14608  14682#  14685  14721#  14724  14843#  14846
MSGJ     14186#  14188
MSGM1    14223#  14226
MSGM10   14320#  14323
MSGM11   14350#  14353
MSGM3    14275#  14277
MULT      134#
NEWTST    134#  1236  1249  1263  1275  1289  1304  1318  1335  1351  1368  1390  1412  1430  1448
         1467  1486  1503  1526  1549  1566  1588  1614  1644  1663  1681  1697  1715  1734  1755
         1775  1796  1817  1839  1863  1886  1909  1930  1946  1965  1983  2006  2023  2038  2056
         2073  2098  2123  2142  2166  2189  2205  2223  2249  2277  2295  2313  2331  2348  2372
         2388  2406  2425  2457  2494  2531  2568  2605  2630  2655  2680  2706  2732  2758  2784
         2810  2832  2855  2891  2927  2945  2963  2981  3004  3020  3048  3062  3076  3090  3105
         3120  3135  3149  3164  3179  3194  3209  3224  3239  3253  3268  3283  3298  3313  3327
         3342  3357  3372  3398  3416  3433  3469  3490  3514  3545  3567  3607  3628  3668  3691
         3711  3730  3754  3775  3803  3824  3842  3860  3940  3954  3968  3988  4010  4030  4052
         4066  4081  4096  4111  4126  4140  4155  4170  4185  4211  4232  4258  4275  4327  4344
         4370  4388  4414  4440  4492  4519  4545  4571  4597  4624  4651  4678  4705  4731  4757
         4783  4809  4835  4861  4888  4916  4944  4972  5000  5028  5058  5084  5119  5145  5176
         5202  5232  5257  5283  5310  5336  5361  5387  5413  5439  5465  5491  5517  5542  5568
         5594  5620  5646  5671  5697  5723  5749  5775  5801  5827  5853  5879  5907  5936  5968
         6000  6027  6054  6081  6108  6135  6162  6188  6215  6242  6269  6296  6323  6350  6377
         6404  6431  6458  6485  6512  6539  6565  6591  6617  6643  6669  6696  6718  6744  6765
         6792  6818  6844  6865  6892  6919  6943  6967  6992  7016  7041  7060  7080  7101  7122
         7151  7180  7209  7238  7267  7296  7325  7354  7376  7398  7420  7442  7468  7495  7522
         7549  7578  7607  7640  7669  7696  7723  7750  7777  7804  7833  7862  7890  7917  7944
         7971  8001  8032  8058  8084  8110  8136  8157  8178  8210  8242  8274  8306  8343  8380
         8417  8454  8491  8528  8569  8606  8638  8670  8702  8734  8755  8776  8797  8818  8840
         8861  8888  8916  8936  8955  8979  9003  9024  9051  9078  9105  9132  9154  9176  9198
         9220  9242  9264  9286  9313  9344  9371  9398  9426  9453  9480  9507  9533  9562  9591
         9620  9649  9678  9706  9735  9764  9792  9819  9846  9874  9901  9928  9954  9981  10008
        10035  10064  10093  10123  10153  10183  10217  10247  10277  10306  10325  10346  10368  10390  10412
        10434  10456  10478  10499  10521  10542  10564  10586  10608  10635  10657  10689  10715  10746  10768
        10798  10824  10850  10876  10906  10936  10964  10995  11024  11049  11078  11111  11143  11176  11204
        11238  11254  11281  11300  11319  11338  11357  11398  11435  11472  11532  11592  11616  11634  11652
        11682  11714  11747  11780  11837  11894  11925  11956  11987  12018  12049  12085  12117  12161  12248
        12268  12308  12346  12371  12396  12421  12445  12484  12522  12574  12634  12694  12724  12756  12788
        12811  12834  12856  12887  12924  12959  12994  13020  13047  13074  13104  13144  13174  13189  13214
        13230  13247  13267  13327  13387  13447  13507  13552  13596  13626  13657  13693  13729  13765  13801
        13832  13856  13885  13914  13943  13965  14011  14035  14060  14084  14112  14141  14186  14224  14275
        14321  14351  14438  14499  14569  14606  14683  14722  14844  14931
NOINST    570#  11257  11341  12997  13023  13050  13077  13107  13299  13359  13419  13479  13525  13569  13599
        13629  13660  13696  13732  13768  13804  13835  13859  13888  13917  13946  14087  14115  14144
NOSCOP    570#  1239  1252  1266  1278  1292  1307  1321  1338  1354  1371  1393  1415  1433  1451
         1470  1489  1506  1529  1552  1569  1591  1617  1647  1666  1684  1700  1718  1737  1758
         1778  1799  1820  1842  1866  1889  1912  1933  1949  1968  1986  2009  2026  2041  2059
         2076  2101  2126  2145  2169  2192  2208  2226  2252  2280  2298  2316  2334  2351  2375
         2391  2409  2428  2460  2497  2534  2571  2608  2633  2658  2683  2709  2735  2761  2787
         2813  2835  2858  2894  2930  2948  2966  2984  3007  3023  3051  3065  3079  3093  3108
         3123  3138  3152  3167  3182  3197  3212  3227  3242  3256  3271  3286  3301  3316  3330
         3345  3360  3375  3401  3419  3436  3472  3493  3517  3548  3570  3610  3631  3671  3694
         3714  3733  3757  3778  3806  3827  3845  3871  13150
```

```
POP       134#  15043  15044  15614  15615
PREERR  15289#  15305
PRENEW    570#  14193  14236  14287  14331  14368  14446  14509  14576  14617  14693  14734  14859  14934
PRESCO  15207#  15223
PUSH      134#  15021  15027  15575  15577  15598
REPORT    134#
SCOPE      29#   3619   3944   3958   3972   3992   4014   4034   4056   4070   4085   4100   4115   4130   4144
          4159   4174   4189   4215   4236   4262   4279   4331   4348   4374   4392   4418   4444   4496   4523
          4549   4575   4601   4628   4655   4682   4709   4735   4761   4787   4813   4839   4865   4892   4920
          4948   4976   5004   5032   5062   5088   5123   5149   5180   5206   5236   5261   5287   5314   5340
          5365   5391   5417   5443   5469   5495   5521   5546   5572   5598   5624   5650   5675   5701   5727
          5753   5779   5805   5831   5857   5883   5911   5940   5972   6004   6031   6058   6085   6112   6139
          6166   6192   6219   6246   6273   6300   6327   6354   6381   6408   6435   6462   6489   6516   6543
          6569   6595   6621   6647   6673   6700   6722   6748   6769   6796   6822   6848   6869   6896   6923
          6947   6971   6996   7020   7045   7064   7084   7105   7126   7155   7184   7213   7242   7271   7300
          7329   7358   7380   7402   7424   7446   7472   7499   7526   7553   7582   7611   7644   7673   7700
          7727   7754   7781   7808   7837   7866   7894   7921   7948   7975   8005   8036   8062   8088   8114
          8140   8161   8182   8214   8246   8278   8310   8347   8384   8421   8458   8495   8532   8573   8610
          8642   8674   8706   8738   8759   8780   8801   8822   8844   8865   8892   8920   8940   8959   8983
          9007   9028   9055   9082   9109   9136   9158   9180   9202   9224   9246   9268   9290   9317   9348
          9375   9402   9430   9457   9484   9511   9537   9566   9595   9624   9653   9682   9710   9739   9768
          9796   9823   9850   9878   9905   9932   9958   9985  10012  10039  10068  10097  10127  10157  10187
         10221  10251  10281  10310  10329  10350  10372  10394  10416  10438  10460  10482  10503  10525  10546
         10568  10590  10612  10639  10661  10693  10719  10750  10772  10802  10828  10854  10880  10910  10940
         10968  10999  11028  11053  11082  11115  11147  11180  11208  11242  11258  11285  11304  11323  11342
         11361  11402  11439  11476  11536  11596  11620  11638  11656  11686  11718  11751  11784  11841  11898
         11929  11960  11991  12022  12053  12089  12124  12169  12252  12272  12312  12350  12375  12400  12425
         12449  12488  12526  12578  12638  12698  12728  12760  12792  12815  12838  12860  12893  12928  12963
         12998  13024  13051  13078  13108  13135  13178  13193  13216  13234  13251  13300  13360  13420  13480
         13526  13570  13600  13630  13661  13697  13733  13769  13805  13836  13860  13889  13918  13947  13973
         14015  14039  14064  14088  14116  14145  14195  14238  14289  14333  14370  14448  14511  14578  14619
         14695  14736  14861  14936  14982
SETPRI    134#
SETTRA  15649#  15658  15659  15660
SETUP     134#
SKIP      134#   1243   1258   1270   1284   1299   1312   1331   1347   1363   1385   1407   1425   1443   1462
          1481   1499   1521   1544   1561   1583   1609   1638   1658   1676   1692   1710   1729   1750   1771
          1791   1812   1834   1857   1880   1903   1941   1960   1978   2001   2018   2051   2068   2093   2118
          2137   2162   2184   2201   2218   2244   2272   2290   2308   2326   2343   2383   2401   2420   2452
          2488   2525   2562   2599   2626   2651   2676   2701   2727   2753   2779   2805   2828   2850   2941
          2959   2977   2999   3015   3042   3057   3070   3084   3100   3115   3129   3144   3158   3173   3189
          3204   3219   3234   3247   3263   3278   3293   3307   3321   3337   3352   3367   3393   3411   3428
          3464   3485   3509   3540   3561   3601   3662   3706   3726   3770   3816   3836   3854   3950   3963
          3984   4006   4026   4048   4062   4076   4091   4106   4122   4136   4150   4165   4180   4207   4228
          4254   4271   4323   4340   4366   4384   4410   4436   4488   4515   4541   4567   4593   4620   4647
          4674   4701   4727   4753   4779   4805   4831   4857   4884   4911   4939   4967   4995   5023   5053
          5079   5114   5140   5171   5197   5227   5252   5279   5306   5332   5357   5383   5409   5435   5461
          5487   5513   5538   5564   5590   5616   5642   5667   5693   5719   5745   5771   5797   5823   5849
          5875   5902   5931   5963   5995   6022   6049   6076   6103   6130   6157   6183   6210   6237   6264
          6291   6318   6345   6372   6399   6426   6453   6480   6507   6534   6561   6587   6613   6639   6665
          6691   6713   6739   6760   6787   6813   6839   6860   6888   6915   6939   6963   6988   7012   7037
          7056   7076   7096   7117   7147   7176   7205   7234   7263   7292   7321   7350   7371   7393   7415
          7437   7464   7491   7518   7545   7573   7602   7635   7664   7692   7719   7746   7773   7800   7828
          7857   7885   7913   7940   7967   7996   8027   8054   8080   8106   8132   8152   8173   8204   8236
          8268   8300   8337   8374   8411   8448   8485   8522   8563   8600   8632   8664   8696   8728   8750
          8771   8792   8813   8834   8856   8884   8912   8932   8951   8975   8999   9019   9047   9073   9100
```

```
                9127    9149    9171    9193    9215    9237    9259    9281    9309    9340    9367    9394    9421    9449    9476
                9503    9529    9557    9586    9615    9644    9673    9702    9730    9759    9787    9815    9842    9870    9897
                9924    9951    9977   10004   10031   10059   10088   10118   10148   10178   10212   10242   10272   10301   10321
               10341   10363   10385   10407   10429   10451   10473   10494   10516   10537   10559   10577   10582   10599   10604
               10621   10631   10648   10653   10670   10680   10683   10702   10707   10710   10728   10732   10742   10759   10764
               10781   10791   10794   10811   10816   10819   10837   10842   10845   10863   10868   10871   10889   10892   10897
               10900   10919   10922   10927   10930   10955   11068   11097   11133   11162   11195   11224   11249   11275   11278
               11296   11315   11334   11353   11390   11427   11464   11491   11499   11511   11520   11551   11559   11571   11580
               11629   11647   13479   12517   12556   12625   12685   12910   12945   12980   13543   13587   13622   13653   13689
               13725   13761   13797   13828   13852   13881   13910   13939   13961   14004   14032   14057   14081   14109   14138
               14170
SLASH           134#
SPACE           134#
STARS           134#    148     150     157     170     180     233     236    1236    1238    1249    1251    1263    1265    1275
              1277    1289    1291    1304    1306    1318    1320    1335    1337    1351    1353    1368    1370    1390    1392
              1412    1414    1430    1432    1448    1450    1467    1469    1486    1488    1503    1505    1526    1528    1549
              1551    1566    1568    1588    1590    1614    1616    1644    1646    1663    1665    1681    1683    1697    1699
              1715    1717    1734    1736    1755    1757    1775    1777    1796    1798    1817    1819    1839    1841    1863
              1865    1886    1888    1909    1911    1930    1932    1946    1948    1965    1967    1983    1985    2006    2008
              2023    2025    2038    2040    2056    2058    2073    2075    2098    2100    2123    2125    2142    2144    2166
              2168    2189    2191    2205    2207    2223    2225    2249    2251    2277    2279    2295    2297    2313    2315
              2331    2333    2348    2350    2372    2374    2388    2390    2406    2408    2425    2427    2457    2459    2494
              2496    2531    2533    2568    2570    2605    2607    2630    2632    2655    2657    2680    2682    2706    2708
              2732    2734    2758    2760    2784    2786    2810    2812    2832    2834    2855    2857    2891    2893    2927
              2929    2945    2947    2963    2965    2981    2983    3004    3006    3020    3022    3048    3050    3062    3064
              3076    3078    3090    3092    3105    3107    3120    3122    3135    3137    3149    3151    3164    3166    3179
              3181    3194    3196    3209    3211    3224    3226    3239    3241    3253    3255    3268    3270    3283    3285
              3298    3300    3313    3315    3327    3329    3342    3344    3357    3359    3372    3374    3398    3400    3416
              3418    3433    3435    3469    3471    3490    3492    3514    3516    3545    3547    3567    3569    3607    3609
              3628    3630    3668    3670    3691    3693    3711    3713    3730    3732    3754    3756    3775    3777    3803
              3805    3824    3826    3842    3844    3860    3870    3940    3942    3954    3956    3968    3970    3988    3990
              4010    4012    4030    4032    4052    4054    4066    4068    4081    4083    4096    4098    4111    4113    4126
              4128    4140    4142    4155    4157    4170    4172    4185    4187    4211    4213    4232    4234    4258    4260
              4275    4277    4327    4329    4344    4346    4370    4372    4388    4390    4414    4416    4440    4442    4492
              4494    4519    4521    4545    4547    4571    4573    4597    4599    4624    4626    4651    4653    4678    4680
              4705    4707    4731    4733    4757    4759    4783    4785    4809    4811    4835    4837    4861    4863    4888
              4890    4916    4918    4944    4946    4972    4974    5000    5002    5028    5030    5058    5060    5084    5086
              5119    5121    5145    5147    5176    5178    5202    5204    5232    5234    5257    5259    5283    5285    5310
              5312    5336    5338    5361    5363    5387    5389    5413    5415    5439    5441    5465    5467    5491    5493
              5517    5519    5542    5544    5568    5570    5594    5596    5620    5622    5646    5648    5671    5673    5697
              5699    5723    5725    5749    5751    5775    5777    5801    5803    5827    5829    5853    5855    5879    5881
              5907    5909    5936    5938    5968    5970    6000    6002    6027    6029    6054    6056    6081    6083    6108
              6110    6135    6137    6162    6164    6188    6190    6215    6217    6242    6244    6269    6271    6296    6298
              6323    6325    6350    6352    6377    6379    6404    6406    6431    6433    6458    6460    6485    6487    6512
              6514    6539    6541    6565    6567    6591    6593    6617    6619    6643    6645    6669    6671    6696    6698
              6718    6720    6744    6746    6765    6767    6792    6794    6818    6820    6844    6846    6865    6867    6892
              6894    6919    6921    6943    6945    6967    6969    6992    6994    7016    7018    7041    7043    7060    7062
              7080    7082    7101    7103    7122    7124    7151    7153    7180    7182    7209    7211    7238    7240    7267
              7269    7296    7298    7325    7327    7354    7356    7376    7378    7398    7400    7420    7422    7442    7444
              7468    7470    7495    7497    7522    7524    7549    7551    7578    7580    7607    7609    7640    7642    7669
              7671    7696    7698    7723    7725    7750    7752    7777    7779    7804    7806    7833    7835    7862    7864
              7890    7892    7917    7919    7944    7946    7971    7973    8001    8003    8032    8034    8058    8060    8084
              8086    8110    8112    8136    8138    8157    8159    8178    8180    8210    8212    8242    8244    8274    8276
              8306    8308    8343    8345    8380    8382    8417    8419    8454    8456    8491    8493    8528    8530    8569
              8571    8606    8608    8638    8640    8670    8672    8702    8704    8734    8736    8755    8757    8776    8778
              8797    8799    8818    8820    8840    8842    8861    8863    8888    8890    8916    8918    8936    8938    8955
```

	8957	8979	8981	9003	9005	9024	9026	9051	9053	9078	9080	9105	9107	9132	9134
	9154	9156	9176	9178	9198	9200	9220	9222	9242	9244	9264	9266	9286	9288	9313
	9315	9344	9346	9371	9373	9398	9400	9426	9428	9453	9455	9480	9482	9507	9509
	9533	9535	9562	9564	9591	9593	9620	9622	9649	9651	9678	9680	9706	9708	9735
	9737	9764	9766	9792	9794	9819	9821	9846	9848	9874	9876	9901	9903	9928	9930
	9954	9956	9981	9983	10008	10010	10035	10037	10064	10066	10093	10095	10123	10125	10153
	10155	10183	10185	10217	10219	10247	10249	10277	10279	10306	10308	10325	10327	10346	10348
	10368	10370	10390	10392	10412	10414	10434	10436	10456	10458	10478	10480	10499	10501	10521
	10523	10542	10544	10564	10566	10586	10588	10608	10610	10635	10637	10657	10659	10689	10691
	10715	10717	10746	10748	10768	10770	10798	10800	10824	10826	10850	10852	10876	10878	10906
	10908	10936	10938	10964	10966	10995	10997	11024	11026	11049	11051	11078	11080	11111	11113
	11143	11145	11176	11178	11204	11206	11238	11240	11254	11256	11281	11283	11300	11302	11319
	11321	11338	11340	11357	11359	11398	11400	11435	11437	11472	11474	11532	11534	11592	11594
	11616	11618	11634	11636	11652	11654	11682	11684	11714	11716	11747	11749	11780	11782	11837
	11839	11894	11896	11925	11927	11956	11958	11987	11989	12018	12020	12049	12051	12085	12087
	12117	12122	12161	12167	12248	12250	12268	12270	12308	12310	12346	12348	12371	12 3	12396
	12398	12421	12423	12445	12447	12484	12486	12522	12524	12574	12576	12634	12636	12 4	12696
	12724	12726	12756	12758	12788	12790	12811	12813	12834	12836	12856	12858	12887	12891	12924
	12926	12959	12961	12994	12996	13020	13022	13047	13049	13074	13076	13104	13106	13144	13149
	13174	13176	13189	13191	13214	13216	13230	13232	13247	13249	13267	13298	13327	13358	13387
	13418	13447	13478	13507	13524	13552	13568	13596	13598	13626	13628	13657	13659	13693	13695
	13729	13731	13765	13767	13801	13803	13832	13834	13856	13858	13885	13887	13914	13916	13943
	13945	13965	13971	14011	14013	14035	14037	14060	14062	14084	14086	14112	14114	14141	14143
	14186	14192	14224	14235	14275	14286	14321	14330	14351	14367	14438	14445	14499	14508	14569
	14575	14606	14616	14683	14692	14722	14733	14844	14858	14931	14933	14976	15017	15033	15077
	15211	15293	15356	15412	15492	15570	15628								
SWRSU	1348														
TESTNO	5708	1240	1253	1267	1279	1293	1308	1322	1339	1355	1372	1394	1416	1434	1452
	1471	1490	1507	1530	1553	1570	1592	1618	1648	1667	1685	1701	1719	1738	1759
	1779	1800	1821	1843	1867	1890	1913	1934	1950	1969	1987	2010	2027	2042	2060
	2077	2102	2127	2146	2170	2193	2209	2227	2253	2281	2299	2317	2335	2352	2376
	2392	2410	2429	2461	2498	2535	2572	2609	2634	2659	2684	2710	2736	2762	2788
	2814	2836	2859	2895	2931	2949	2967	2985	3008	3024	3052	3066	3080	3094	3109
	3124	3139	3153	3168	3183	3198	3213	3228	3243	3257	3272	3287	3302	3317	3331
	3346	3361	3376	3402	3420	3437	3473	3494	3518	3549	3571	3611	3632	3672	3695
	3715	3734	3758	3779	3807	3828	3846	3872	3945	3959	3973	3993	4015	4035	4057
	4071	4086	4101	4116	4131	4145	4160	4175	4190	4216	4237	4263	4280	4332	4349
	4375	4393	4419	4445	4497	4524	4550	4576	4602	4629	4656	4683	4710	4736	4762
	4788	4814	4840	4866	4893	4921	4949	4977	5005	5033	5063	5089	5124	5150	5181
	5207	5237	5262	5288	5315	5341	5366	5392	5418	5444	5470	5496	5522	5547	5573
	5599	5625	5651	5676	5702	5728	5754	5780	5806	5832	5858	5884	5912	5941	5973
	6005	6032	6059	6086	6113	6140	6167	6193	6220	6247	6274	6301	6328	6355	6382
	6409	6436	6463	6490	6517	6544	6570	6596	6622	6648	6674	6701	6723	6749	6770
	6797	6823	6849	6870	6897	6924	6948	6972	6997	7021	7046	7065	7085	7106	7127
	7156	7185	7214	7243	7272	7301	7330	7359	7381	7403	7425	7447	7473	7500	7527
	7554	7583	7612	7645	7674	7701	7728	7755	7782	7809	7838	7867	7895	7922	7949
	7976	8006	8037	8063	8089	8115	8141	8162	8183	8215	8247	8279	8311	8348	8385
	8422	8459	8496	8533	8574	8611	8643	8675	8707	8739	8760	8781	8802	8823	8845
	8866	8893	8921	8941	8960	8984	9008	9029	9056	9083	9110	9137	9159	9181	9203
	9225	9247	9269	9291	9318	9349	9376	9403	9431	9458	9485	9512	9538	9567	9596
	9625	9654	9683	9711	9740	9769	9797	9824	9851	9879	9906	9933	9959	9986	10013
	10040	10069	10098	10128	10158	10188	10222	10252	10282	10311	10330	10351	10373	10395	10417
	10439	10461	10483	10504	10526	10547	10569	10591	10613	10640	10662	10694	10720	10751	10773
	10803	10829	10855	10881	10911	10941	10969	11000	11029	11054	11083	11116	11148	11181	11209
	11243	11259	11286	11305	11324	11343	11362	11403	11440	11477	11537	11597	11621	11639	11657
	11687	11719	11752	11785	11842	11899	11930	11961	11992	12023	12054	12090	12125	12170	12253

```
                12273   12313   12351   12376   12401   12426   12450   12489   12527   12579   12639   12699   12729   12761   12793
                12816   12839   12861   12894   12929   12964   12999   13025   13052   13079   13109   13151   13179   13194   13219
                13235   13252   13301   13361   13421   13481   13527   13571   13601   13631   13662   13698   13734   13770   13806
                13837   13861   13890   13919   13948   13974   14016   14040   14065   14089   14117   14146   14194   14237   14288
                14332   14369   14447   14510   14577   14618   14694   14735   14860   14935
TRMTRP          15649#
TYPBIN            134#
TYPDEC            134#
TYPNAM            134#
TYPNUM            134#
TYPOCS            134#
TYPOCT            134#  15368   15392
TYPTXT            134#
UPCODE          15015#  15040
YESCOP            570#   3943    3957    3971    3991    4013    4033    4055    4069    4084    4099    4114    4129    4143    4158
                 4173    4188    4214    4235    4261    4278    4330    4347    4373    4391    4417    4443    4495    4522    4548
                 4574    4600    4627    4654    4681    4708    4734    4760    4786    4812    4838    4864    4891    4919    4947
                 4975    5003    5031    5061    5087    5122    5148    5179    5205    5235    5260    5286    5313    5339    5364
                 5390    5416    5442    5468    5494    5520    5545    5571    5597    5623    5649    5674    5700    5726    5752
                 5778    5804    5830    5856    5882    5910    5939    5971    6003    6030    6057    6084    6111    6138    6165
                 6191    6218    6245    6272    6299    6326    6353    6380    6407    6434    6461    6488    6515    6542    6568
                 6594    6620    6646    6672    6699    6721    6747    6768    6795    6821    6847    6868    6895    6922    6946
                 6970    6995    7019    7044    7063    7083    7104    7125    7154    7183    7212    7241    7270    7299    7328
                 7357    7379    7401    7423    7445    7471    7498    7525    7552    7581    7610    7643    7672    7699    7726
                 7753    7780    7807    7836    7865    7893    7920    7947    7974    8004    8035    8061    8087    8113    8139
                 8160    8181    8213    8245    8277    8309    8346    8383    8420    8457    8494    8531    8572    8609    8641
                 8673    8705    8737    8758    8779    8800    8821    8843    8864    8891    8919    8939    8958    8982    9006
                 9027    9054    9081    9108    9135    9157    9179    9201    9223    9245    9267    9289    9316    9347    9374
                 9401    9429    9456    9483    9510    9536    9565    9594    9623    9652    9681    9709    9738    9767    9795
                 9822    9849    9877    9904    9931    9957    9984   10011   10038   10067   10096   10126   10156   10186   10220
                10250   10280   10309   10328   10349   10371   10393   10415   10437   10459   10481   10502   10524   10545   10567
                10589   10611   10638   10660   10692   10718   10749   10771   10801   10827   10853   10879   10909   10939   10967
                10998   11027   11052   11081   11114   11146   11179   11207   11241   11284   11303   11322   11360   11401   11438
                11475   11535   11595   11619   11637   11655   11685   11717   11750   11783   11840   11897   11928   11959   11990
                12021   12052   12088   12123   12168   12251   12271   12311   12349   12374   12399   12424   12448   12487   12525
                12577   12637   12697   12727   12759   12791   12814   12837   12859   12892   12927   12962   13177   13192   13217
                13233   13250   13972   14014   14038   14063
SSCMRE            178#    217     218     219     220     221     222
SSCMTM            178#    223     224     225     226     227
SSESCA            134#
SSNEWT            134#   1236    1249    1263    1275    1289    1304    1318    1335    1351    1368    1390    1412    1430    1448
                 1467    1486    1503    1526    1549    1566    1588    1614    1644    1663    1681    1697    1715    1734    1755
                 1775    1796    1817    1839    1863    1886    1909    1930    1946    1965    1983    2006    2023    2038    2056
                 2073    2098    2123    2142    2166    2189    2205    2223    2249    2277    2295    2313    2331    2348    2372
                 2388    2406    2425    2457    2494    2531    2568    2605    2630    2655    2680    2706    2732    2758    2784
                 2810    2832    2855    2891    2927    2945    2963    2981    3004    3020    3048    3062    3076    3090    3105
                 3120    3135    3149    3164    3179    3194    3209    3224    3239    3253    3268    3283    3298    3313    3327
                 3342    3357    3372    3398    3416    3433    3469    3490    3514    3545    3567    3607    3628    3668    3691
                 3711    3730    3754    3775    3803    3824    3842    3860    3940    3954    3968    3988    4010    4030    4052
                 4066    4081    4096    4111    4126    4140    4155    4170    4185    4211    4232    4258    4275    4327    4344
                 4370    4388    4414    4440    4492    4519    4545    4571    4597    4624    4651    4678    4705    4731    4757
                 4783    4809    4835    4861    4888    4916    4944    4972    5000    5028    5058    5084    5119    5145    5176
                 5202    5232    5257    5283    5310    5336    5361    5387    5413    5439    5465    5491    5517    5542    5568
                 5594    5620    5646    5671    5697    5723    5749    5775    5801    5827    5853    5879    5907    5936    5968
                 6000    6027    6054    6081    6108    6135    6162    6188    6215    6242    6269    6296    6323    6350    6377
                 6404    6431    6458    6485    6512    6539    6565    6591    6617    6643    6669    6696    6718    6744    6765
```

```
                6792    6818    6844    6865    6892    6919    6943    6967    6992    7016    7041    7060    7080    7101    7122
                7151    7180    7209    7238    7267    7296    7325    7354    7376    7398    7420    7442    7468    7495    7522
                7549    7578    7607    7640    7669    7696    7723    7750    7777    7804    7833    7862    7890    7917    7944
                7971    8001    8032    8058    8084    8110    8136    8157    8178    8210    8242    8274    8306    8343    8380
                8417    8454    8491    8528    8569    8606    8638    8670    8702    8734    8755    8776    8797    8818    8840
                8861    8888    8916    8936    8955    8979    9003    9024    9051    9078    9105    9132    9154    9176    9198
                9220    9242    9264    9286    9313    9344    9371    9398    9426    9453    9480    9507    9533    9562    9591
                9620    9649    9678    9706    9735    9764    9792    9819    9846    9874    9901    9928    9954    9981   10008
               10035   10064   10093   10123   10153   10183   10217   10247   10277   10306   10325   10346   10368   10390   10412
               10434   10456   10478   10499   10521   10542   10564   10586   10608   10635   10657   10689   10715   10746   10768
               10798   10824   10850   10876   10906   10936   10964   10995   11024   11049   11078   11111   11143   11176   11204
               11238   11254   11281   11300   11319   11338   11357   11398   11435   11472   11532   11592   11616   11634   11652
               11682   11714   11747   11780   11837   11894   11925   11956   11987   12018   12049   12085   12117   12161   12248
               12268   12308   12346   12371   12396   12421   12445   12484   12522   12574   12634   12694   12724   12756   12788
               12811   12834   12856   12887   12924   12959   12994   13020   13047   13074   13104   13144   13174   13189   13214
               13230   13247   13267   13327   13387   13447   13507   13552   13596   13626   13657   13693   13729   13765   13801
               13832   13856   13885   13914   13943   13965   14011   14035   14060   14084   14112   14141   14186   14224   14275
               14321   14351   14438   14499   14569   14606   14683   14722   14844   14931
$$SET         15649#  15658   15659   15660
$$SKIP          134#  1244    1258    1271    1284    1299    1313    1331    1347    1363    1385    1407    1425    1443    1462
                1481    1499    1521    1544    1561    1583    1609    1638    1658    1676    1692    1710    1729    1750    1771
                1791    1812    1834    1857    1880    1903    1941    1960    1978    2001    2018    2051    2068    2093    2118
                2137    2162    2184    2201    2218    2244    2272    2290    2308    2326    2343    2383    2401    2420    2452
                2488    2525    2562    2599    2626    2651    2676    2701    2727    2753    2779    2805    2828    2850    2941
                2959    2977    2999    3015    3043    3057    3071    3085    3100    3115    3130    3144    3159    3174    3189
                3204    3219    3234    3248    3263    3278    3293    3308    3322    3337    3352    3367    3393    3411    3428
                3464    3485    3509    3540    3561    3601    3662    3706    3726    3770    3816    3836    3854    3950    3964
                3984    4006    4026    4048    4062    4077    4092    4107    4122    4136    4151    4166    4181    4207    4228
                4254    4271    4323    4340    4366    4384    4410    4436    4488    4515    4541    4567    4593    4620    4647
                4674    4701    4727    4753    4779    4805    4831    4857    4884    4911    4939    4967    4995    5023    5053
                5079    5114    5140    5171    5197    5227    5252    5279    5306    5332    5357    5383    5409    5435    5461
                5487    5513    5538    5564    5590    5616    5642    5667    5693    5719    5745    5771    5797    5823    5849
                5875    5902    5931    5963    5995    6022    6049    6076    6103    6130    6157    6183    6210    6237    6264
                6291    6318    6345    6372    6399    6426    6453    6480    6507    6534    6561    6587    6613    6639    6665
                6691    6713    6739    6760    6787    6813    6839    6860    6888    6915    6939    6963    6988    7012    7037
                7056    7076    7096    7117    7147    7176    7205    7234    7263    7292    7321    7350    7371    7393    7415
                7437    7464    7491    7518    7545    7573    7602    7635    7664    7692    7719    7746    7773    7800    7828
                7857    7885    7913    7940    7967    7996    8027    8054    8080    8106    8132    8152    8173    8204    8236
                8268    8300    8337    8374    8411    8448    8485    8522    8563    8600    8632    8664    8696    8728    8750
                8771    8792    8813    8834    8856    8884    8912    8932    8951    8975    8999    9019    9047    9073    9100
                9127    9149    9171    9193    9215    9237    9259    9281    9309    9340    9367    9394    9421    9449    9476
                9503    9529    9557    9586    9615    9644    9673    9702    9730    9759    9787    9815    9842    9870    9897
                9924    9951    9977   10004   10031   10059   10088   10118   10148   10178   10212   10242   10272   10301   10321
               10341   10363   10385   10407   10429   10451   10473   10494   10516   10537   10559   10577   10582   10599   10604
               10621   10631   10648   10653   10670   10680   10683   10702   10707   10710   10728   10732   10742   10759   10764
               10781   10791   10794   10811   10816   10819   10837   10842   10845   10863   10868   10871   10889   10892   10897
               10900   10919   10922   10927   10930   10955   11068   11097   11133   11162   11195   11224   11249   11275   11278
               11296   11315   11334   11353   11390   11427   11464   11491   11499   11511   11520   11551   1.559   11571   11580
               11629   11647   12479   12517   12556   12625   12685   12910   12945   12980   13543   13587   13622   13653   13689
               13725   13761   13797   13828   13852   13881   13910   13939   13961   14004   14032   14057   14081   14109   14138
               14170
.EQUAT          1#     24
.HEADE          1#      2
.SWRHI          1#     12
.SWRLO         23#
.$ACT1          1#    168
```

```
.SAPTB      1#    234#
.SAPTH      1#    146
.SAPTY      1#  15568
.SCATC      1#    134
.SCMTA      1#    178
.SEOP       1#  14974
.SERRO      1#  15291
.SERRT      1#  15354
.SPOWE      1#  15015
.SSCOP      1#  15209
.STRAP      1#  15626
.STYPD      1#
.STYPE      1#  15410
.STYPO      1#  15490


. ABS.  067770     000

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 DSKZ:DQKDAA,DSKZ:DQKDAA/CRF/SOL/DS:ERFZ=DQKDAA
 RUN-TIME: 61 64 5 SECONDS
 RUN-TIME RATIO: 349/132=2.6
 CORE USED:  28K  (56 PAGES)
```