KD11-K

KD11-K BLT
CQKDAD0

AH-8092D-MC

FICHE 1 OF 2

NOV 1979
digital
MADE IN USA

KD11-K

KD11 K BLT
CQKDAD0

AH 8092D MC

FICHE 2 OF 2

NOV 1979
digital
MADE IN USA

```
                        .SBTTL  DOCUMENT LISTING
                        .TITLE  CQKDAD0 KD11-K BLT
                        .REM %
```

PRODUCT CODE: AC-8090D-MC

PRODUCT NAME: CQKDAD0 KD11-K BLT

PRODUCT DATE: JULY 1979

MAINTAINER: DIAGNOSTIC ENGINEERING

TABLE OF CONTENTS

```
                        1.0     GENERAL PROGRAM INFORMATION
                                ---------------------------

                        1.1     PROGRAM PURPOSE
                                ---------------

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

                                A. BASIC CPU TESTS (BCPT)
                                   ----------------------

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

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

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

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

                                C. COMPREHENSIVE INSTRUCTION TESTS (CIT)
                                   -------------------------------------

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

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

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

D. COMBINED INSTRUCTION EXERCISER (IEX)

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

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

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

B. SOFTWARE REQUIREMENTS

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

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCHY REQUIREMENTS

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

1.5 FAILURE ASSUMPTIONS

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

2.0 OPERATING INSTRUCTIONS

```
                        ----------------------

                2.1     LOADING AND STARTING PROCEDURES
                        -------------------------------

                        A. LOADING PROCEDURES

                                1) STANDARD PDP11 ABSOLUTE LOADER PROCEDURES FOR PAPER TAPE.

                                2) STANDARD XXDP MONITOR LOADING PROCEDURES.

                                3) STANDARD APT OR ACT LOADING

                        B. MANUAL STARTING PROCEDURES

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


                2.2     SPECIAL ENVIRONMENTS
                        --------------------

                        16K PDP11/6X SERIES SYSTEMS

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

                2.3     PROGRAM OPTIONS
                        ---------------

                        A. SWITCH REGISTER OPTIONS

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

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

                        SW14    CONTINUOUSLY LOOP ON THE CURRENT TEST

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

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

```
252
253                                     SW11    INHIBIT SUB-TEST ITERATIONS. TEST ITERATIONS ARE
254                                             AUTOMATICALLY INHIBITED ON THE FIRST PASS.
255
256                                     SW10    SEARCH FOR AND CONTINUOUSLY LOOP ON THE TEST NUMBER
257                                             SELECTED BY THE CONTENTS OF SW<08:00>. ONLY USE THIS
258                                             OPTION FOR TESTS TST176 THRU TST767 SINCE THE "SCOPE"
259                                             UTILITY IS NOT ACTIVE UNTIL TEST TST176.  LOOPING ON
260                                             TST176 WILL CAUSE A LOOP ON THE ENTIRE "BIT" SECTION
261                                             (TESTS 0-176).
262
263                                     SW09    LOCK ON HARD ERROR
264
265                                     SW<8:0> USED TO SELECT A PARTICULAR TEST FOR LOOPING IF SW10-1.
266                                             TEST NUMBER MUST BE BETWEEN 176 AND 767.
267
268                                     B. MEMORY LOCATIONS
269
270                                     4. BPTLOC: THERE IS A LOCATION TAGGED "BPTLOC" THAT PROVIDES THE
271                                                USER THE MECHANISM FOR SETTING SIXTEEN "BREAKPOINT
272                                                HALTS" THROUGHOUT THE PROGRAM. THIS ENABLES RAPIDLY
273                                                "HOMING IN" ON THE FAILING TEST IN THOSE CASES WHERE
274                                                THE FAULT CAUSES A RUNAWAY OR HUNG PROGRAM. REFER TO
275                                                PARA. 4.2 FOR A DETAILED DESCRIPTION OF THE USE OF
276                                                THIS FEATURE.
277
278                             2.4     EXECUTION TIMES
279                                     ---------------
280
281                                     ONE COMPLETE ERROR FREE PASS OF "CQKDA" WITH NO TEST ITERATIONS
282                                     SHOULD TAKE LESS THAN 7 SECONDS. A SUCCESSFUL PASS WILL BE IN-
283                                     DICATED BY THE FOLLOWING PRINTOUT ON THE CONSOLE DEVICE:
284
285                                             END PASS # 000001       ERROR COUNT = 000000
286
287                                     THIS ERROR COUNT IS NOT CLEARED AT THE BEGINNING OF A NEW PASS.
288                                     WITH ITERATIONS ENABLED A COMPLETE ERROR FREE PASS SHOULD TAKE
289                                     LESS THAN 2.5 MINUTES.
290
291                             3.0     ERROR INFORMATION
292                                     -----------------
293
294                             3.1     ERROR REPORTING PROCEDURES
295                                     --------------------------
296
297                                     A. ERROR MESSAGE FORMATS
298
299                                     THERE ARE SEVERAL DIFFERENT ERROR FORMATS. EACH IS DESCRIBED BELOW.
300
301                                     1.) ERROR 1 IS OF THE FORM
302
303                             S/B DST WAS DST  DEST    (IR)    TEST    (PC)    (SP)   (PSW)
304                             XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX
305
306                                     WHERE:
307
```

```
                        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
```

```
364                                     5.) OTHER ERRORS TYPE OUT THEIR SPECIFIC ERROR MESSAGE, FOLLOWED
365                                     BY SELF EXPLANATORY DATA HEADERS, DEPENDING ON THE ERROR.
366                                     AN EXAMPLE FOLLOWS:
367
368                                 BAD DATA READ BY A MED
369                                   PC    MEDCODE EXPECTD RECEIVD
370                                 XXXXXX  XXXXXX  XXXXXX  XXXXXX
371
372                                     6.) WHEN THE SCOPE ROUTINE BECOMES ACTIVE, IT CHECKS THAT
373                                     THE TEST NUMBER (IN R0) IS EXACTLY ONE GREATER THAN THE
374                                     TEST NUMBER ON THE PREVIOUS SCOPE CALL.  IF A MACHINE
375                                     ERROR CAUSES TESTS TO BE SKIPPED, OR THE PROGRAM TO JUMP
376                                     BACKWARDS, ERROR 11 WILL REPORT THIS AS FOLLOWS:
377
378                                 TESTS SKIPPED
379                                   PC    EXPCTD  ACTUAL  (TEST #'S)
380                                 XXXXXX  XXXXXX  XXXXXX
381
382                                     EXPCTD  THIS IS THE TEST NUMBER THE SCOPE WAS EXPECTING TO BE
383                                             CALLED FROM.
384
385                                     ACTUAL  THIS IS THE TEST NUMBER THAT IT FOUND IN R0
386
387                                     7.) RESERVED INSTRUCTION TRAP ERROR MESSAGE
388                                         ---------------------------------------
389
390                                         ANY RESERVED INSTRUCTION TRAP DETECTED AFTER THE
391                                         BASIC TESTS RESULTS IN THE FOLLOWING PRINTOUT:
392
393                                         TRAPPED TO 10  PC = XXXXXX
394
395                                         WHERE:  XXXXXX IS THE VALUE OF THE PROGRAM COUNTER
396                                                  PUSHED ON THE STACK WHEN THE TRAP WAS SPRUNG.
397
398                                         AFTER REPORTING THE ERROR, THE PROGRAM IS RESTARTED
399                                         FROM THE BEGINNING.
400
401                                         IF A RSVD INSTRUCTION TRAP OCCURS WHILE IN THE PROCESS
402                                         OF TRYING TO SERVICE A PREVIOUS RSVD INSTRUCTION TRAP
403                                         OR A BUS ERROR TRAP THE PROGRAM HALTS. A DESCRIPTION
404                                         OF THIS HALT IS CONTAINED IN PARA. 3.2.3 BELOW.
405
406                                         IF A RSVD INSTRUCTION TRAP OCCURS PRIOR TO COMPLETION
407                                         OF THE BASIC INSTRUCTION TEST SECTION THE PROGRAM WILL
408                                         HALT VIA A TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF
409                                         THIS HALT IS DESCRIBED IN PARA. 3.2.2 BELOW.
410
411
412                                 4. BUS ERROR TRAP ERROR MESSAGE
413                                    ---------------------------
414
415                                         ANY UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD
416                                         ADDRESS ERROR, ILLEGAL INSTRUCTION, OR STACK OVERFLOW)
417                                         RESULTS IN THE FOLLOWING PRINTOUT:
418
419                                         TRAPPED TO 4  PC = XXXXXX
```

```
420
421                                     WHERE:  XXXXXX IS THE VALUE OF THE PC PUSHED ONTO
422                                             THE STACK WHEN THE TRAP WAS SPRUNG.
423
424                                     AFTER REPORTING THE ERROR THE PROGRAM IS RESTARTED
425                                     FROM THE BEGINNING.
426
427                                     IF A BUS ERROR TRAP OCCURS WHILE A PREVIOUS BUS ERROR
428                                     OR RSVD INSTRUCTION IS STILL PENDING THE PROGRAM WILL
429                                     HALT. A DESCRIPTION OF THE HALT INTERPRETATION IS GIVEN
430                                     IN PARA. 3.2.3 BELOW.
431
432                                     IF A BUS ERROR OCCURS PRIOR TO THE COMPLETION OF THE
433                                     BASIC INSTRUCTION TESTS, THE PROGRAM WILL HALT VIA A
434                                     TRAPCATCHER IN THE VECTOR. A DESCRIPTION OF THIS HALT
435                                     IS INCLUDED IN PARA. 3.2.2 BELOW.
436
437                                5. POWER FAIL
438                                   -----------
439
440                                   IF A POWER FAIL CONDITION IS DETECTED, THE FOLLOWING
441                                   MESSAGE IS PRINTED:
442
443                                         POWER
444
445                                   AFTER PRINTING AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
446                                   THE BEGINNING.
447
448                        3.2     ERROR HALTS
449                                ----------
450
451                                1. BASIC INSTRUCTION TESTS (BIT)
452                                   ----------------------------
453
454                                     ANY ERROR DETECTED IN THE BASIC TESTS CAUSES THE
455                                     PROGRAM TO HALT WITH THE PC+2 OF THE LOCATION CONTAINING
456                                     THE HALT INSTRUCTION DISPLAYED.
457
458                                     EXAMINING THE CONTENTS OF THE CPU'S GENERAL REGISTERS,
459                                     THE PSW, AND THE STACK WILL PROVIDE ADDITIONAL FAULT
460                                     IDENTIFICATION INFORMATION.
461
462                                     DEPRESSING "CONTINUE" AFTER THE HALT WILL CAUSE AN
463                                     AUTOMATIC RETRY OF THE FAILING TEST. IF THE ERROR IS
464                                     SOLID THE PROGRAM WILL LOCK ON THIS TEST, BUT IF IT
465                                     IS INTERMITTENT THE PROGRAM WILL CONTINUE ON IN NORMAL
466                                     SEQUENCE ONCE THE TEST IS SUCCESSFULLY EXECUTED.
467
468                                     TO ESTABLISH A TIGHT SCOPE LOOP ON THE FAILING TEST,
469                                     REPLACE THE "HALT" WITH A 400(8). AND DEPRESS "CONTINUE"
470                                     THE "400" IS A "BR .+2" WHICH FUNCTIONS AS A NOP. THIS
471                                     IS NECESSARY TO PRESERVE THE INTEGRITY OF THE CONDITION
472                                     CODE OPERATE INSTRUCTION THAT IS USED AS A SCOPE SYNC. THIS
473                                     BUILT IN SYNC FEATURE IS DESCRIBED IN PARA. 5.0.
474
475                                2. TRAPCATCHER HALTS
```

```
                                                                K  1
                                        ------------------
                                          THE VECTOR AREA (LOC 000 - 776) IS PROGRAM LOADED WITH
                                          A STANDARD TRAPCATCHER AS SHOWN BELOW:

                                                  V   / V+2
                                                  V+2/ HALT

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

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

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

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

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

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

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

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

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

                                          THE STACK PROVIDES THE FOLLOWING ADDITIONAL INFORMATION:
```

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

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

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

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

                                    PASCNT = XXXXXX  ERRCNT = YYYYYY

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

                                            YYYYYY IS THE TOTAL ERROR COUNT IN OCTAL

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

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

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

                            B. PROGRAM IDENTIFICATION MESSAGE AS DESCRIBED BELOW:

                                    CQKDA_  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:
```

588			
589		[BPTLOC]	ADDRESS DISPLAY
590			HALT PC+2
591			
592		BIT00=1	4326
593		BIT01=1	6312
594		BIT02=1	10632
595		BIT03=1	11762
596		BIT04=1	14356
597		BIT05=1	17116
598		BIT06=1	21542
599		BIT07=1	24350
600		BIT08=1	27162
601		BIT09-1	32156
602		BIT10-1	34642
603		BIT11=1	37452
604		BIT12=1	42142
605		BIT13=1	46142
606		BIT14=1	52602
607		BIT15=1	55426
608			
609		NOTE: IF THE USER DEPOSITED A 000400(8) IN LOCATION "BPTLOC"	
610		ONLY ONE HALT WOULD OCCUR AND AT THAT TIME THE	
611		DISPLAY SHOULD CONTAIN 27162.	
612			
613		THIS FEATURE IS USEFUL FOR TRACKING DOWN THE TEST THAT CAUSES	
614		A "RUNAWAY" OR "HUNG" PROGRAM.	
615			
616		LOCATION "BPTLOC" IS PROGRAM LOADED AS 000000 TO INHIBIT ANY HALTS.	
617			
618			
619	5.0	MAINTENANCE PROCEDURES	
620		----------	
621			
622	5.1	THE KD11-K PROCESSOR	
623		----------	
624			
625		THE PROCEDURES OUTLINED IN THIS SECTION ASSUME THAT "CQKDA"	
626		CAN BE LOADED INTO CORE AND STARTED. IF THE FAILURE MODE	
627		PREVENTS PROGRAM LOADING OR AFFECTS NORMAL POWER UP AND	
628		CONSOLE OPERATIONS, THE TECHNICIAN MUST REVERT TO THE MANUAL	
629		DEBUG AND CHECKOUT PROCEDURES.	
630			
631		THE KD11-K CENTRAL PROCESSING UNIT CAN BE VIEWED AS	
632		CONSISTING OF TWO MAJOR LOGIC AREAS AS DEPICTED BELOW:	
633			
634			
635		DATA IN ****	
636		*	
637		*	
638		V	
639		********** **********	
640		* * * *	
641		* CONTROL *>>>>>>>>>>>>>>* DATA *	
642		* SECTION *<<<<<<<<<<<<<<* PATHS *	
643		* * * *	

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

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

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

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

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

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

                        FROM THE BIT SECTION TO THE MED TESTS IN THE CIT SECTION, ALL
                        TEST INSTRUCTIONS ARE PRECEEDED BY A CONDITION CODE OPERATE
                        INSTRUCTION.  THE UBREAK REGISTER IS PROGRAM LOADED TO GENERATE
                        A SYNC PULSE NEAR THE END OF THIS INSTRUCTION.  DURING THE MED
                        TESTS, THE PULSE IS GENERATED NEAR THE BEGINNING OF THE MED
                        EXECUTION.  THIS PULSE IS GENERATED ON BACKPLANE PIN B03M2 AND
                        MAY BE USED IN CONJUNTION WITH THE PROGRAM LOOPING FEATURES TO
                        PROBE THE KD11-K DURING THE FAILING TEST.



                6.0     ********** ECO HISTORY **********
                        HISTORY STARTS WITH REV. D0
                        CQKDAD0 - HARDWARE ECO #M7877-002 REQUIRES THAT BIT 9 IN LOG JAM REG. BE
                                  TESTED FOR SET CONDITION. TEST 763 MODIFIED.


                %

                .TITLE  CQKDA-D KD11-K BASIC LOGIC TESTS
                ;*COPYRIGHT (C) 1977,1979
                ;*DIGITAL EQUIPMENT CORP.
                ;*MAYNARD, MASS. 01754
```

```
                        ;*
                        ;*
                        ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                        ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
                        ;*
                        .SBTTL  OPERATIONAL SWITCH SETTINGS
                        ;*
                        ;*      SWITCH                  USE
                        ;*      ------          --------------------
                        ;*        15            HALT ON ERROR
                        ;*        14            LOOP ON TEST
                        ;*        13            INHIBIT ERROR TYPEOUTS
                        ;*        12            INHIBIT ID MESSAGE & UNEXPECTED TRAP MESSAGES
                        ;*        11            INHIBIT ITERATIONS
                        ;*        10            LOOP ON TEST IN SWR<8:0>
                        ;*         9            LOOP ON ERROR
                        .ENABLE ABS
                        .SBTTL  BASIC DEFINITIONS

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

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

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

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

```
                                ;*"SWITCH REGISTER" SWITCH DEFINITIONS
000000 100000                   SW15=   100000
000000 040000                   SW14=   40000
000000 020000                   SW13=   20000
000000 010000                   SW12=   10000
000000 004000                   SW11=   4000
000000 002000                   SW10=   2000
000000 001000                   SW09=   1000
000000 000400                   SW08=   400
000000 000200                   SW07=   200
000000 000100                   SW06=   100
000000 000040                   SW05=   40
000000 000020                   SW04=   20
000000 000010                   SW03=   10
000000 000004                   SW02=   4
000000 000002                   SW01=   2
000000 000001                   SW00=   1
                                .EQUIV  SW09,SW9
                                .EQUIV  SW08,SW8
                                .EQUIV  SW07,SW7
                                .EQUIV  SW06,SW6
                                .EQUIV  SW05,SW5
                                .EQUIV  SW04,SW4
                                .EQUIV  SW03,SW3
                                .EQUIV  SW02,SW2
                                .EQUIV  SW01,SW1
                                .EQUIV  SW00,SW0

                                ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
000000 100000                   BIT15=  100000
000000 040000                   BIT14=  40000
000000 020000                   BIT13=  20000
000000 010000                   BIT12=  10000
000000 004000                   BIT11=  4000
000000 002000                   BIT10=  2000
000000 001000                   BIT09=  1000
000000 000400                   BIT08=  400
000000 000200                   BIT07=  200
000000 000100                   BIT06=  100
000000 000040                   BIT05=  40
000000 000020                   BIT04=  20
000000 000010                   BIT03=  10
000000 000004                   BIT02=  4
000000 000002                   BIT01=  2
000000 000001                   BIT00=  1
                                .EQUIV  BIT09,BIT9
                                .EQUIV  BIT08,BIT8
                                .EQUIV  BIT07,BIT7
                                .EQUIV  BIT06,BIT6
                                .EQUIV  BIT05,BIT5
                                .EQUIV  BIT04,BIT4
                                .EQUIV  BIT03,BIT3
                                .EQUIV  BIT02,BIT2
                                .EQUIV  BIT01,BIT1
```

```
                                        .EQUIV  BIT00,BIT0

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

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

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

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

                                        ;;**************************************************************
                                        ;HOOKS REQUIRED BY ACT11
            000714                              $SVPC=.                 ;SAVE PC
            000046                              .=46
000046  060644                              $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
```

```
                000052                  .=52
000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
                000714                  .=$SVPC                 ;; RESTORE PC
```

```
                                .SBTTL  COMMON TAGS

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

001000                                  .=1000
001000                  $CMTAG:                         ;;START OF COMMON TAGS
001000  000000                  .WORD   0
001002  000             $TSTNM: .BYTE   0               ;;CONTAINS THE TEST NUMBER
001003  000             $ERFLG: .BYTE   0               ;;CONTAINS ERROR FLAG
001004  000000          $ICNT:  .WORD   0               ;;CONTAINS SUBTEST ITERATION COUNT
001006  000000          $LPADR: .WORD   0               ;;CONTAINS SCOPE LOOP ADDRESS
001010  000000          $LPERR: .WORD   0               ;;CONTAINS SCOPE RETURN FOR ERRORS
001012  000000          $ERTTL: .WORD   0               ;;CONTAINS TOTAL ERRORS DETECTED
001014  000             $ITEMB: .BYTE   0               ;;CONTAINS ITEM CONTROL BYTE
001015  001             $ERMAX: .BYTE   1               ;;CONTAINS MAX. ERRORS PER TEST
001016  000000          $ERRPC: .WORD   0               ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001020  000000          $GDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'GOOD' DATA
001022  000000          $BDADR: .WORD   0               ;;CONTAINS ADDRESS OF 'BAD' DATA
001024  000000          $GDDAT: .WORD   0               ;;CONTAINS 'GOOD' DATA
001026  000000          $BDDAT: .WORD   0               ;;CONTAINS 'BAD' DATA
001030  000000                  .WORD   0               ;;RESERVED--NOT TO BE USED
001032  000000                  .WORD   0
001034  000             $AUTOB: .BYTE   0               ;;AUTOMATIC MODE INDICATOR
001035  000             $INTAG: .BYTE   0               ;;INTERRUPT MODE INDICATOR
001036  000000                  .WORD   0
001040  177570          SWR:    .WORD   DSWR            ;;ADDRESS OF SWITCH REGISTER
001042  177570          DISPLAY: .WORD  DDISP           ;;ADDRESS OF DISPLAY REGISTER
001044  177560          $TKS:   177560                  ;;TTY KBD STATUS
001046  177562          $TKB:   177562                  ;;TTY KBD BUFFER
001050  177564          $TPS:   177564                  ;;TTY PRINTER STATUS REG. ADDRESS
001052  177566          $TPB:   177566                  ;;TTY PRINTER BUFFER REG. ADDRESS
001054  000             $NULL:  .BYTE   0               ;;CONTAINS NULL CHARACTER FOR FILLS
001055  002             $FILLS: .BYTE   2               ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001056  012             $FILLC: .BYTE   12              ;;INSERT FILL CHARS. AFTER A "LINE FEED"
001057  000             $TPFLG: .BYTE   0               ;;"TERMINAL AVAILABLE" FLAG (BIT<07>-0=YES)
001060  000000          $REGAD: .WORD   0               ;;CONTAINS THE ADDRESS FROM
                                                        ;;WHICH  ($REG0) WAS OBTAINED
001062  000000          $REG0:  .WORD   0               ;;CONTAINS (($REGAD)+0)
001064  000000          $REG1:  .WORD   0               ;;CONTAINS (($REGAD)+2)
001066  000000          $REG2:  .WORD   0               ;;CONTAINS (($REGAD)+4)
001070  000000          $REG3:  .WORD   0               ;;CONTAINS (($REGAD)+6)
001072  000000          $REG4:  .WORD   0               ;;CONTAINS (($REGAD)+10)
001074  000000          $REG5:  .WORD   0               ;;CONTAINS (($REGAD)+12)
001076  000000          $TMP0:  .WORD   0               ;;USER DEFINED
001100  000000          $TMP1:  .WORD   0               ;;USER DEFINED
001102  000000          $TMP2:  .WORD   0               ;;USER DEFINED
001104  000000          $TMP3:  .WORD   0               ;;USER DEFINED
001106  000000          $TMP4:  .WORD   0               ;;USER DEFINED
001110  000000          $TIMES: 0                       ;;MAX. NUMBER OF ITERATIONS
001112  000000          $ESCAPE:0                       ;;ESCAPE ON ERROR ADDRESS
001114  077             $QUES:  .ASCII  /?/             ;;QUESTION MARK
001115  015             $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
001116  000012          $LF:    .ASCIZ  <12>            ;;LINE FEED
                                ;;*****************************************************************
```

```
                                .SBTTL  APT MAILBOX-ETABLE

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

```
                                .SBTTL  ERROR POINTER TABLE

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

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


001150                          $ERRTB:

                                ;ITEM 1
001150  064640                          EM1     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001152  000000                          0
001154  067764                          DT1     ;$REG4,  $REG3,   $REG2, $REG1,$REG0,$ERRPC,$REG5,$REG6
001156  000000                          0
                                ;ITEM 2
001160  064640                          EM2     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001162  065061                          DH2     ;                       IS R3
001164  067764                          DT2     ;$REG4,  $REG3,  $REG2, $REG1, $REG0,$ERRPC,$REG5,$REG6
001166  000000                          0
                                ;ITEM 3
001170  065013                          EM3     ;S/B SP  WAS SP  (IR)  TEST  (PC)  (PSW)
001172  000000                          0
001174  070006                          DT3     ;$REG4,  $REG3,  $REG1,$REG0,$ERRPC,$REG6
001176  000000                          0
                                ;ITEM 4
001200  064640                          EM4     ;S/B DST  WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001202  065072                          DH4     ;                       IS R5
001204  067764                          DT4     ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001206  000000                          0
                                ;ITEM 5
001210  064666                          EM5     ;(IR)  TEST  (PC)  (SP)  (PSW)
001212  000000                          0
001214  067772                          DT5     ;$REG1, $REG0, $ERRPC, $REG5, $REG6
001216  000000                          0
                                ;ITEM 6
001220  064660                          EM6     ;  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001222  000000                          0
001224  067770                          DT6     ;$REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001226  000000                          0
                                ;ITEM 7
001230  064650                          EM7     ;WAS DST  DEST  (IR)  TEST  (PC)  (SP)  (PSW)
001232  000000                          0
001234  067766                          DT7     ;$REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001236  000000                          0
                                ;ITEM 10
001240  064724                          EM10    ;S/B RES  WAS RES  DST OP  STC OP  TEST  (PC) (SP) (PSW)
001242  000000                          0
001244  067764                          DT10    ;$REG4, $REG3, $REG2, $REG1, $REG0, $ERRPC, $REG5, $REG6
001246  000000                          0
```

```
1010                                    ;ITEM 11
1011  001250  065261                    EM11    ;TESTS SKIPPED
1012  001252  065277                    DH11    ;  PC    EXPCTD ACTUAL (TEST #'S)
1013  001254  070024                    DT11    ;$ERRPC, $TESTN,$REG0
1014  001256  000000                    0
1015
1016                                    ;ITEM 12
1017  001260  065335                    EM12    ;MED DID NOT ABORT IN USER MODE
1018  001262  067373                    DH23    ;PC
1019  001264  067664                    DT23    ;$ERRPC
1020  001266  000000                    0
1021
1022                                    ;ITEM 13
1023  001270  065374                    EM13    ;MED EXECUTED IN USER MODE
1024  001272  067373                    DH23    ;PC
1025  001274  067664                    DT23    ;$ERRPC
1026  001276  000000                    0
1027
1028                                    ;ITEM 14
1029  001300  065426                    EM14    ;MED CHANGED PSW
1030  001302  067373                    DH23    ;PC
1031  001304  067664                    DT23    ;$ERRPC
1032  001306  000000                    0
1033
1034                                    ;ITEM 15
1035  001310  065446                    EM15    ;MICROBREAK TRAP-TO-4 DID NOT OCCUR
1036  001312  067304                    DH15    ;ERRPC MEDCODE MICROBK REG.
1037  001314  067630                    DT15    ;$ERRPC,$TMP0,$TMP1,0
1038  001316  067756                    DF15    ;0,0
1039
1040                                    ;ITEM 16
1041
1042  001320  067035                    EM16    ;CACHE DATA LOGGED INCORRECTLY
1043  001322  067607                    DH44    ;PC EXPCT RECVD
1044  001324  067670                    DT24    ;$ERRPC,$REG1,$REG0,0
1045  001326  000000                    0
1046
1047                                    ;ITEM 17
1048
1049  001330  067006                    EM45    ;CACHE TAG LOGGED WRONG
1050  001332  067607                    DH44    ;PC EXPCT RECVD
1051  001334  067670                    DT24    ;$ERRPC,$REG0,$REG1,0
1052  001336  000000                    0
1053
1054                                    ;ITEM 20
1055
1056  001340  065705                    EM26    ;PHYS. BA LOGGED WRONG
1057  001342  067607                    DH44    ;PC EXPCT RECVD
1058  001344  067670                    DT24    ;$ERRPC,$REG1,$REG0,0
1059  001346  000000                    0
1060
1061                                    ;ITEM 21
1062  001350  065535                    EM21    ;CSP CONSTANT WRONG
1063  001352  067336                    DH17    ;PC MEDCODE EXPECTD RECEIVD
1064  001354  067640                    DT21    ;$ERRPC,$TMP1,$TMP2,$REG0,0
1065  001356  067760                    DF17    ;0,0,0
```

```
1066
1067                                    ;ITEM 22
1068  001360  065560                   EM22    ;BAD DATA READ BY A MED
1069  001362  067336                   DH17    ;PC MEDCODE EXPECTD RECEIVD
1070  001364  067652                   DT22    ;$ERRPC,$TMP1,$TMP2,$TMP3,0
1071  001366  067760                   DF17    ;0,0,0
1072
1073                                    ;ITEM 23
1074  001370  065607                   EM23    ;NO ODD PC TRAP
1075  001372  067373                   DH23    ;PC
1076  001374  067664                   DT23    ;$ERRPC
1077  001376  000000                   0
1078
1079                                    ;ITEM 24
1080
1081  001400  065626                   EM24    ;ODD ADR. BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1082  001402  067400                   DH24    ;PC CPUERR LOGJAM
1083  001404  067670                   DT24    ;$ERRPC,$REG1,$REG0
1084  001406  000000                   0
1085
1086                                    ;ITEM 25
1087
1088  001410  065511                   EM17    ;LOG CUA LOGGED INCORRECT U-ADDR
1089  001412  067607                   DH44    ;PC     EXPCTD  RECVD
1090  001414  067670                   DT24    ;$ERRPC $REG1   $REG0
1091  001416  000000                   0
1092
1093                                    ;ITEM 26
1094
1095  001420  065705                   EM26    ;PHYS. BA LOGGED WRONG
1096  001422  067437                   DH26    ;PC PA<17:16>-EXPCT-PA<15:0>  PA<17:16>-RECVD-PA<15:0>
1097  001424  067706                   DT26    ;$ERRPC,$REG1,$REG2,$REG0,$REG3,0
1098  001426  000000                   0
1099
1100                                    ;ITEM 27
1101
1102  001430  065732                   EM27    ;CACHE PARITY ERROR LOGGED IN BACK UP MODE
1103  001432  067521                   DH27    ;PC LOGPBA LOGDATA LOGTAG
1104  001434  067722                   DT27    ;$ERRPC,$REG3,$REG1,$REG2
1105  001436  000000                   0
1106
1107                                    ;ITEM 30
1108
1109  001440  066002                   EM30    ;CACHE PARITY TRAPPED WHEN DISABLED
1110  001442  067373                   DH23    ;PC
1111  001444  067664                   DT23    ;$ERRPC
1112  001446  000000                   0
1113
1114                                    ;ITEM 31
1115
1116  001450  066605                   EM31    ;NO CACHE PARITY TRAP
1117  001452  067373                   DH23    ;PC
1118  001454  067664                   DT23    ;$ERRPC
1119  001456  000000                   0
1120
1121                                    ;ITEM 32
```

```
1122
1123  001460  066114                  EM32    ;MEMORY ERROR REGISTERS INCORRECT
1124  001462  067554                  DH32    ;PC MEMERR
1125  001464  067700                  DT25    ;$ERRPC,$REG0
1126  001466  000000                  0
1127
1128                                  ;ITEM 33
1129
1130  001470  066145                  EM33    ;TIMEOUT BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1131  001472  067400                  DH24    ;PC CPUERR LOGJAM
1132  001474  067670                  DT24    ;$ERRPC,$REG1,$REG0
1133  001476  000000                  0
1134
1135                                  ;ITEM 34
1136
1137  001500  066223                  EM34    ;NO ILLEGAL INTERNAL ADDRESS TRAP
1138  001502  067373                  DH23    ;PC
1139  001504  067664                  DT23    ;$ERRPC
1140  001506  000000                  0
1141
1142                                  ;ITEM 35
1143
1144  001510  066260                  EM35    ;INTERNAL ADDRESS ERROR BIT NOT SET IN CPU ERROR REGISTER OR LOG JAM
1145  001512  067400                  DH24    ;PC CPUERR LOGJAM
1146  001514  067670                  DT24    ;$ERRPC,$REG1,$REG0
1147  001516  000000                  0
1148
1149                                  ;ITEM 36
1150
1151  001520  066346                  EM36    ;LAST INTERRUPT/TRAP VECTOR NOT LOGGED IN FLAG REGISTER
1152  001522  067423                  DH25    ;PC FLGREG
1153  001524  067700                  DT25    ;$ERRPC,$REG0
1154  001526  000000                  0
1155
1156                                  ;ITEM 37
1157
1158  001530  066423                  EM37    ;LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR
1159  001532  067400                  DH24    ;PC CPUERR LOGJAM
1160  001534  067670                  DT24    ;$ERRPC,$REG1,$REG0
1161  001536  000000                  0
1162
1163                                  ;ITEM 40
1164
1165  001540  066516                  EM40    ;ERROR LOG WAS NOT RE-ENABLED, ODD ADR BIT CLR IN CPUERR
1166  001542  067400                  DH24    ;PC CPUERR LOGJAM
1167  001544  067670                  DT24    ;$ERRPC,$REG1,$REG0
1168  001546  000000                  0
1169
1170                                  ;ITEM 41
1171
1172  001550  066045                  EM41    ;INSTRUCTION NOT ABORTED IN CACHE ABORT MODE
1173  001552  067373                  DH23    ;PC
1174  001554  067664                  DT23    ;$ERRPC
1175  001556  000000                  0
1176
1177                                  ;ITEM 42
```

```
1178
1179  001560  066632                EM42    ;LO BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE
1180  001562  067570                DH42    ;PC LOGSERVCE
1181  001564  067700                DT25    ;$ERRPC,$REG0,0
1182  001566  000000                0
1183
1184                                ;ITEM 43
1185
1186  001570  066720                EM43    ;LO BYTE & TAG PARITY BITS NOT SET IN MEM ERR REGISTER
1187  001572  067554                DH32    ;PC MEMERR
1188  001574  067700                DT25    ;$ERRPC,$REG0
1189  001576  000000                0
1190
1191                                ;ITEM 44
1192
1193  001600  067065                EMEIS1  ;EIS SET COND CODES WRONG
1194  001602  067222                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1195  001604  067734                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1196  001606  000000                0
1197
1198                                ;ITEM 45
1199
1200  001610  067116                EMEIS2  ;EIS GAVE WRONG RESULT
1201  001612  067222                DHEIS1  ; PSW   REG-WAS-REG+1   REG-S/B-REG+1     PC      TEST    (IR)
1202  001614  067734                DTEIS1  ;$REGAD $REG2   $REG3   $REG1   $REG4   $ERRPC  $REG0   $TMP0
1203  001616  000000                0
1204
1205                                ;ITEM 46
1206
1207  001620  067144                EM46    ;AUTO-INCREMENT (DECREMENT) DID NOT OCCUR
1208  001622  067263                DH46    ;  PC    (IR)    TEST
1209  001624  067746                DT46    ;$ERRPC $TMP0   $REG0
1210  001626  000000                0
1211
1212          076600                        MED = 076600
1213          140000                        UM=     140000
1214          177770                        UBREAK= 177770
1215          177744                        MEMERR=177744
1216          177766                        CPUERR=177766
1217          177746                        CCR=177746
1218          000100                        WWP=BIT6
1219          000001                        DPTRP=BIT0
1220          000200                        PABORT=BIT7
1221          000100                        LO=BIT6
1222          000200                        HI-BIT7
1223          000040                        TAG=BIT5
1224
1225                                        .EQUIV  SP,KSP
1226
1227
1228                                ;*      MED OPERATION CODE DEFINITIONS
1229
1230          000226                        WCNSSW=226
1231          000022                        RDWHAMI=022
1232          000222                        WRWHAMI=222
1233          000144                        RDFLAG=144
```

```
1234    000344          WRFLAG=344
1235    000100          RDLJAM=100
1236    000300          WRLJAM=300
1237    000101          RDLSERVICE=101
1238    000301          WRLSERVICE=301
1239    000102          RDLPBA=102
1240    000302          WRLPBA=302
1241    000103          RDLCUA=103
1242    000303          WRLCUA=303
1243    000104          RDLFGINT=104
1244    000304          WRLFGINT=304
1245    000105          RDLWHAMI=105
1246    000305          WRLWHAMI=305
1247    000106          RDLDATA=106
1248    000306          WRLDATA=306
1249    000107          RDLTAG=107
1250    000307          WRLTAG=307
1251    000071          SWB01=71                ;MICRO ADDR. IN SWAB INST.
1252
1253
1254
1255                    ;ADDRESS ASSIGNMENTS FOR DL11 CONSOLE TERMINAL INTERFACE
1256
1257    177560          RCSR=177560             ;RCVR. CONTROL / STATUS REG. ADDRESS
1258    177562          RDBR = 177562           ;RECEIVER DATA BUFFER REG. ADDR.
1259    177564          XCSR = 177564           ;TRANSMITTER CONTROL / STATUS REG. ADDR
1260    177566          XDBR = 177566           ;TRANSMIT DATA BUFFER REG. ADDR.
1261    177546          LKCSR=  177546          ;LINE CLOCK ADDRESS
1262
```

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

```
1317
1318                    ; ******************************************************************
1319                            .SBTTL BT005 "CCC AND COND. BR'S" TEST - FLAGS CLEARED
1320                    ; ******************************************************************
1321
```

```
1322  001700  000257          BT005:  CCC                     ;MAKE N:C=0000
1323
1324  001702  100403          1005:   BMI     E005            ;BR IF "N" STILL SET
1325  001704  001402                  BEQ     E005            ;BR IF "Z" STILL SET
1326  001706  102401                  BVS     E005            ;BR IF "V" STILL SET
1327  001710  103002                  BCC     BT006           ;BR IF "C" GOT CLEARED
1328
1329  001712  000000          E005:   HALT                    ;ERROR - ONE OF THE ABOVE BR'S FAILED
1330                                                          ;OR THE CCC FAILED TO CLEAR ALL FLAGS
1331  001714  000771                  BR      BT005           ;LOCK ON HARD ERROR
1332
1333                          ; ***************************************************************
1334                                  .SBTTL BT006 "CLR  %R" TEST - SETS THE "Z" BIT
1335                          ; ***************************************************************
1336
1337  001716  000257          BT006:  CCC                     ;MAKE N:C=0000
1338
1339  001720  005000          1006:   CLR     R0              ;TEST THE CLR - IT SHOULD SET "Z"
1340
1341  001722  001402                  BEQ     BT007           ;BR IF CLR SET "Z"
1342
1343  001724  000000          E006:   HALT                    ;ERROR - CLR FAILED TO SET "Z"
1344  001726  000773                  BR      BT006           ;LOCK ON HARD ERROR
1345
1346                          ; ***************************************************************
1347                                  .SBTTL BT007 "TST  %R" TEST - USING THE CLR
1348                          ; ***************************************************************
1349
1350  001730  005000          BT007:  CLR     R0              ;MAKE [R0] = 000000
1351  001732  000257                  CCC                     ;MAKE N:C=0000
1352
1353  001734  005700          1007:   TST     R0              ;TEST THE TST - IT SHOULD SET "Z"
1354
1355  001736  001402                  BEQ     BT010           ;BR IF "Z" SET OK
1356
1357  001740  000000          E007:   HALT                    ;ERROR - CLR FAILED TO LOAD R0 WITH
1358                                                          ;ALL ZEROES OR TST FAILED
1359  001742  000772                  BR      BT007           ;LOCK ON HARD ERROR
1360
1361                          ; ***************************************************************
1362                                  .SBTTL BT010 "COM  %R" TEST - SHOULD SET "N" AND "C"
1363                          ; ***************************************************************
1364
1365  001744  005000          BT010:  CLR     R0              ;MAKE [R0] = 000000
1366  001746  000257                  CCC                     ;MAKE N:C=0000
1367
1368  001750  005100          1010:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
1369
1370  001752  100001                  BPL     E010            ;BR IF "N" FAILED TO SET
1371  001754  103402                  BCS     BT011           ;BR IF "C" SET OK
1372
1373  001756  000000          E010:   HALT                    ;ERROR - COM FAILED
1374  001760  000771                  BR      BT010           ;LOCK ON HARD ERROR
1375
1376                          ; ***************************************************************
1377                                  .SBTTL BT011 "COM  %R AND ADC  %R" TEST
```

```
 1378                                   ; ***************************************************************
 1379
 1380  001762  005000                   BT011:  CLR     R0              ;MAKE [R0] = 000000
 1381  001764  000257                           CCC                     ;MAKE N:C=0000
 1382
 1383  001766  005100                   I011:   COM     R0              ;TEST THE COM - [R0] S/B = 177777
 1384  001770  005500                           ADC     R0              ;TEST THE ADC - [R0] S/B = 000000
 1385
 1386  001772  001001                           BNE     E011            ;BR IF "Z" DID NOT SET
 1387  001774  103402                           BCS     BT012           ;BR IF "C" SET OK
 1388
 1389  001776  000000                   E011:   HALT                    ;ERROR - COM OR ADC FAILED
 1390  002000  000770                           BR      BT011           ;LOCK ON HARD ERROR
 1391
 1392                                   ; ***************************************************************
 1393                                           .SBTTL BT012 "MOV  #N,R" TEST WITH N=177777,[R]=000000
 1394                                   ; ***************************************************************
 1395
 1396  002002  005000                   BT012:  CLR     R0              ;MAKE [R0] = 000000
 1397  002004  000257                           CCC                     ;MAKE N:C=0000
 1398
 1399  002006  012700  177777           I012:   MOV     #-1,R0          ;TEST THE MOV - [R0] S/B = 177777
 1400
 1401  002012  005100                           COM     R0              ;MAKE [R0] = 000000
 1402  002014  001402                           BEQ     BT013           ;BR IF "Z" SET
 1403
 1404  002016  000000                   E012:   HALT                    ;ERROR - MOV FAILED TO LOAD R0 WITH ALL 1'S
 1405  002020  000770                           BR      BT012           ;LOCK ON HARD ERROR
 1406
 1407                                   ; ***************************************************************
 1408                                           .SBTTL BT013 "MOV  #N,R" TEST WITH N=000000,[R]=177777
 1409                                   ; ***************************************************************
 1410
 1411  002022  005000                   BT013:  CLR     R0              ;MAKE [R0] = 000000
 1412  002024  005100                           COM     R0              ;MAKE [R0] = 177777
 1413  002026  000257                           CCC                     ;SCOPE SYNC
 1414
 1415  002030  012700  000000           I013:   MOV     #0,R0           ;TEST THE MOV - [R0] S/B = 000000
 1416
 1417  002034  005100                           COM     R0              ;MAKE [R0] = 177777, SET "C"
 1418  002036  005500                           ADC     R0              ;MAKE [R0] = 000000
 1419  002040  001402                           BEQ     BT014           ;BR IF "Z" GOT SET
 1420
 1421  002042  000000                   E013:   HALT                    ;ERROR - MOV FAILED TO CLEAR R0
 1422  002044  000766                           BR      BT013           ;LOCK ON HARD ERROR
 1423
 1424                                   ; ***************************************************************
 1425                                           .SBTTL BT014 "CLR  (R)" TEST - [R] = 177776
 1426                                   ; ***************************************************************
 1427
 1428  002046  012706  001000           BT014:  MOV     #STACK,SP       ;SET UP STACK POINTER
 1429  002052  012700  177776                   MOV     #PSW,R0         ;R0 POINTS TO PSW
 1430  002056  000277                           SCC                     ;MAKE [PSW] = 017
 1431
 1432  002060  005010                   I014:   CLR     (R0)            ;TEST THE CLR - IT SHOULD CLEAR PSW
 1433
```

```
1434  002062  001002                  BNE     BT015           ;BR IF CLR MADE "Z" = 0 - IT SHOULD
1435
1436  002064  000000                  E014:   HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
1437  002066  000767                  BR      BT014           ;LOCK ON HARD ERROR
1438
1439                                  ; ***************************************************************
1440                                          .SBTTL BT015 "CLR  (R)+" TEST - [R] = 177776
1441                                  ; ***************************************************************
1442
1443  002070  012700  177776          BT015:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1444  002074  000277                  SCC                     ;MAKE [PSW] = 017
1445
1446  002076  005020                  I015:   CLR     (R0)+           ;TEST THE CLR - IT SHOULD CLEAR PSW
1447
1448  002100  001002                  BNE     A015            ;BR IF CLR MADE "Z" = 0 - IT SHOULD
1449
1450  002102  000000                  E1015A: HALT                    ;ERROR- CLR FAILED TO CLEAR PSW
1451  002104  000771                  BR      BT015           ;LOCK ON HARD ERROR
1452
1453  002106  005700                  A015:   TST     R0              ;AUTO INC SHOULD ZERO R0
1454
1455  002110  001402                  BEQ     BT016           ;BR IF IT DID
1456
1457  002112  000000                  E2015:  HALT                    ;ERROR - AUTOINC. FAILED
1458  002114  000765                  BR      BT015           ;LOCK ON HARD ERROR
1459
1460                                  ; ***************************************************************
1461                                          .SBTTL BT016 "COM  (R)" TEST - [R] = 177776
1462                                  ; ***************************************************************
1463
1464  002116  012700  177776          BT016:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1465  002122  000257                  CCC                     ;MAKE [PSW] = 000
1466
1467  002124  005110                  I016:   COM     (R0)            ;TEST THE COM - [PSW] S/B = 357
1468
1469  002126  100003                  BPL     E016            ;N:C-1111 ?
1470  002130  001002                  BNE     E016
1471  002132  102001                  BVC     E016
1472  002134  103403                  BCS     BT017
1473
1474  002136  005010                  E016:   CLR     (R0)            ;GO TO KERNEL MODE
1475  002140  000000                  HALT                    ;ERROR - COM FAILED TO MAKE [PSW]   357
1476  002142  000765                  BR      BT016           ;LOCK ON HARD ERROR
1477
1478                                  ; ***************************************************************
1479                                          .SBTTL BT017 "COM  (R0)+" TEST - [R0] = 177776
1480                                  ; ***************************************************************
1481
1482  002144  012700  177776          BT017:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1483  002150  005010                  CLR     (R0)            ;MAKE [PSW] = 000
1484  002152  000257                  CCC                     ;SCOPE SYNC
1485
1486  002154  005120                  I017:   COM     (R0)+           ;TEST THE COM - [PSW] S/B   357
1487
1488  002156  100003                  BPL     EA017           ;N:C = 1111 ?
1489  002160  001002                  BNE     EA017
```

```
1490  002162  102001                  BVC     EA017
1491  002164  103405                  BCS     A017
1492
1493  002166  012701  177776  EA017:  MOV     #PSW,R1
1494  002172  005011                  CLR     (R1)
1495  002174  000000                  HALT                    ;COM FAILED TO SET ALL FLAGS
1496  002176  000762                  BR      BT017           ;LOCK ON HARD ERROR
1497
1498  002200  005100          A017:   COM     R0              ;SHOULD MAKE [R0] = 177777
1499  002202  005500                  ADC     R0              ;SHOULD MAKE [R0] - 000000
1500  002204  001405                  BEQ     BT020
1501
1502  002206  012701  177776  E2017:  MOV     #PSW,R1
1503  002212  005011                  CLR     (R1)
1504  002214  000000                  HALT                    ;ERROR - COM FAILED TO AUTO INC. R0
1505  002216  000752                  BR      BT017           ;LOCK ON HARD ERROR
1506
1507                          ; ***************************************************************
1508                                  .SBTTL BT020 "MOV  RA,RB" TEST - WITH [RA] 177777,[RB]=000000
1509                          ; ***************************************************************
1510
1511  002220  012700  177776  BT020:  MOV     #PSW,R0
1512  002224  005010                  CLR     (R0)
1513  002226  005000                  CLR     R0              ;MAKE [R0]=000000
1514  002230  005001                  CLR     R1              ;MAKE [R1]=000000
1515  002232  005101                  COM     R1              ;MAKE [R1]=0207777
1516  002234  000257                  CCC                     ;SCOPE SYNC
1517
1518  002236  010100          I020:   MOV'    R1,R0           ;TEST THE MOV
1519
1520  002240  100402                  BMI     A020            ;BR IF "N" GOT SET
1521
1522  002242  000000          EA020:  HALT                    ;ERROR-MOV FAILED TO SET "N"
1523  002244  000765                  BR      BT020           ;LOCK ON HARD ERROR
1524
1525  002246  005100          A020:   COM     R0              ;[R0] SHOULD GO TO 000000
1526  002250  001402                  BEQ     BT021           ;BR IF IT DID
1527
1528  002252  000000          E2020:  HALT                    ;ERROR-MOV FAILED TO LOAD R0 WITH 1'S
1529  002254  000761                  BR      BT020           ;LOCK ON HARD ERROR
1530
1531                          ; ***************************************************************
1532                                  .SBTTL BT021 "MOV  RA,RB" TEST WITH [RA]=000000,[RB] 177777
1533                          ; ***************************************************************
1534
1535  002256  005000          BT021:  CLR     R0              ;MAKE [R0]=000000
1536  002260  005100                  COM     R0              ;MAKE [R0]=177777
1537  002262  005001                  CLR     R1              ;MAKE [R1]=000000
1538  002264  000257                  CCC                     ;SCOPE SYNC
1539
1540  002266  010100          I021:   MOV     R1,R0           ;TEST THE MOV
1541
1542  002270  001402                  BEQ     A021            ;BR IF "Z" GOT SET
1543
1544  002272  000000          EA021:  HALT                    ;MOV FAILED TO SET "Z"
1545  002274  000770                  BR      BT021           ;LOCK ON HARD ERROR
```

```
1546
1547  002276  005100                  A021:   COM     R0              ;SHOULD MAKE [R0]=177777 AND SET "C"
1548  002300  005500                          ADC     R0              ;SHOULD MAKE [R0]=000000
1549  002302  001402                          BEQ     BT022           ;BR IF "Z" SET
1550
1551  002304  000000                  E2021:  HALT                    ;MOV FAILED TO ZERO R0
1552  002306  000763                          BR      BT021           ;LOCK ON HARD ERROR
1553
1554                                  ; ******************************************************************
1555                                          .SBTTL BT022 "MOV  #N,@#A" TEST WITH N=17,A=177776
1556                                  ; ******************************************************************
1557
1558  002310  000257                  BT022:  CCC                     ;MAKE [PSW]=000
1559
1560  002312  012737  000017  177776  1022:   MOV     #17,@#PSW       ;TEST THE MOV
1561
1562  002320  100003                          BPL     E022            ;N:C=1111
1563  002322  001002                          BNE     E022
1564  002324  102001                          BVC     E022
1565  002326  103402                          BCS     BT023
1566
1567  002330  000000                  E022:   HALT                    ;MOV FAILED TO LOAD PSW
1568  002332  000766                          BR      BT022           ;LOCK ON HARD ERROR
1569
1570                                  ; ******************************************************************
1571                                          .SBTTL BT023 "MOV  RA,(RB)+" TEST WITH [RA]=17,[RB]-177776
1572                                  ; ******************************************************************
1573
1574  002334  012700  177776          BT023:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1575  002340  012701  000017                  MOV     #17,R1          ;[SOURCE]=017
1576  002344  000257                          CCC                     ;SCOPE SYNC - MAKE <N:C> = 0000
1577
1578  002346  010120                  1023:   MOV     R1,(R0)+        ;TEST THE MOV
1579
1580  002350  100003                          BPL     EA023           ;N:C = 1111 ?
1581  002352  001002                          BNE     EA023
1582  002354  102001                          BVC     EA023
1583  002356  103402                          BCS     A023
1584
1585  002360  000000                  EA023:  HALT                    ;MOV FAILED TO LOAD PSW
1586  002362  000764                          BR      BT023           ;LOCK ON HARD ERROR
1587
1588  002364  005700                  A023:   TST     R0              ;DID AUTO INC MAKE R0 GO TO 0?
1589  002366  001402                          BEQ     BT024           ;BR IF IT DID
1590
1591  002370  000000                  E2023:  HALT                    ;MOV FAILED TO AUTO INC. R0
1592  002372  000760                          BR      BT023           ;LOCK ON HARD ERROR
1593
1594                                  ; ******************************************************************
1595                                          .SBTTL BT024 "CMP  #N,@#A" TEST WITH N=(A)
1596                                  ; ******************************************************************
1597
1598  002374  012700  177776          BT024:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1599  002400  005010                          CLR     (R0)            ;MAKE [PSW]=000
1600  002402  000273                          273                     ;MAKE N:C=1011
1601
```

```
1602  002404  022737  000013  177776  1024:   CMP     #13,@#PSW       ;TEST THE CMP
1603
1604  002412  001402                          BEQ     BT025           ;BR IF "Z" GOT SET
1605
1606  002414  000000                  E024:   HALT                    ;CMP FAILED TO SET "Z"
1607  002416  000766                          BR      BT024           ;LOCK ON HARD ERROR
1608
1609                                  ; ***************************************************************
1610                                          .SBTTL BT025 "CMP  #N,@#A" WITH N >  (A)
1611                                  ; ***************************************************************
1612
1613  002420  000257                  BT025:  CCC                     ;MAKE [PSW]=000
1614
1615  002422  022737  000017  177776  1025:   CMP     #17,@#PSW       ;TEST THE CMP
1616
1617  002430  001401                          BEQ     E025            ;BR IF "Z" GOT SET
1618  002432  000402                          BR      BT026           ;GO TO NEXT TEST
1619
1620  002434  000000                  E025:   HALT                    ;CMP FAILED TO CLEAR "Z"
1621  002436  000770                          BR      BT025           ;LOCK ON HARD ERROR
1622                                  ; ***************************************************************
1623                                          .SBTTL BT026 "CMP  #N,@#A" WITH N <  (A)
1624                                  ; ***************************************************************
1625
1626  002440  000277                  BT026:  SCC                     ;MAKE [PSW]=017
1627
1628  002442  022737  000000  177776  1026:   CMP     #0,@#PSW        ;TEST THE CMP
1629
1630  002450  001401                          BEQ     E026            ;BR IF "Z" GOT SET
1631  002452  000402                          BR      BT027           ;GO TO NEXT TEST
1632
1633  002454  000000                  E026:   HALT                    ;CMP FAILED TO CLEAR "Z"
1634  002456  000770                          BR      BT026           ;LOCK ON HARD ERROR
1635
1636                                  ; ***************************************************************
1637                                          .SBTTL BT027 "CMP  R,#N" TEST WITH [R]=N
1638                                  ; ***************************************************************
1639
1640  002460  012700  177777          BT027:  MOV     #-1,R0          ;MAKE [R0]=177777
1641  002464  000257                          CCC                     ;N:C=0000
1642
1643  002466  020027  177777          1027:   CMP     R0,#-1          ;TEST THE CMP
1644
1645  002472  001402                          BEQ     BT030           ;BR IF CMP SET "Z"
1646
1647  002474  000000                  E027:   HALT                    ;CMP FAILED
1648  002476  000770                          BR      BT027           ;LOCK ON HARD ERROR
1649
1650                                  ; ***************************************************************
1651                                          .SBTTL BT030 "CMP  R,#N" TEST WITH [R] > N
1652                                  ; ***************************************************************
1653
1654  002500  012700  000001          BT030:  MOV     #1,R0           ;MAKE [R0]=000001
1655  002504  000264                          SEZ                     ;SET THE "Z" BIT
1656
1657  002506  020027  177777          1030:   CMP     R0,#-1          ;TEST THE CMP
```

```
1658
1659  002512  001002                    BNE     BT031           ;BR IF CMP CLEARED "Z"
1660
1661  002514  000000                E030: HALT                    ;CMP FAILED
1662  002516  000770                    BR      BT030           ;LOCK ON HARD ERROR
1663                                ; ******************************************************************
1664                                    .SBTTL BT031 "CMP  R,#N" TEST WITH [R] < N
1665                                ; ******************************************************************
1666
1667  002520  012700  000001       BT031: MOV     #1,R0           ;MAKE [R0] = 000001
1668  002524  000264                    SEZ                     ;SET THE "Z" BIT
1669
1670  002526  020027  000017       I031: CMP     R0,#17          ;TEST THE CMP
1671
1672  002532  001002                    BNE     BT032           ;BR IF CMP CLEARED "Z"
1673
1674  002534  000000                E031: HALT                    ;CMP FAILED TO SET "Z"
1675  002536  000770                    BR      BT031           ;LOCK ON HARD ERROR
1676
1677                                ; ******************************************************************
1678                                    .SBTTL BT032 "CMP  (RA)+,RB" TEST WITH [SOURCE]=[RB]
1679                                ; ******************************************************************
1680
1681  002540  012700  177776       BT032: MOV     #PSW,R0         ;R0 POINTS TO PSW
1682  002544  012737  000340  177776     MOV     #340,@#PSW      ;MAKE [PSW]=340
1683  002552  012701  000340            MOV     #340,R1         ;MAKE [DEST]=340
1684  002556  000257                    CCC                     ;N:C=0000
1685
1686  002560  022001                I032: CMP     (R0)+,R1        ;TEST THE CMP
1687
1688  002562  001402                    BEQ     A032            ;BR IF "Z" GOT SET
1689
1690  002564  000000               EA032: HALT                    ;CMP FAILED TO ACCESS PSW
1691  002566  000764                    BR      BT032           ;LOCK ON HARD ERROR
1692
1693  002570  005700                A032: TST     R0              ;"Z" SHOULD SET
1694  002572  001402                    BEQ     BT033           ;BR IF "Z" SET
1695
1696  002574  000000               E2032: HALT                    ;CMP FAILED TO AUTO INC. R0
1697  002576  000760                    BR      BT032           ;LOCK ON HARD ERROR
1698
1699                                ; ******************************************************************
1700                                    .SBTTL BT033 "CMP  (RA)+,RB" TEST WITH [SOURCE]>[RB]
1701                                ; ******************************************************************
1702
1703  002600  012700  177776       BT033: MOV     #PSW,R0         ;R0 POINTS TO PSW
1704  002604  012737  000340  177776     MOV     #340,@#PSW      ;MAKE [PSW]=340
1705  002612  012701  000330            MOV     #330,R1         ;MAKE [DEST]-330
1706  002616  000264                    SEZ                     ;SET THE "Z" BIT
1707
1708  002620  022001                I033: CMP     (R0)+,R1        ;TEST THE CMP
1709
1710  002622  001002                    BNE     A033            ;BR IF "Z" GOT CLEARED
1711
1712  002624  000000               EA033: HALT                    ;CMP FAILED TO ACCESS PSW
1713  002626  000764                    BR      BT033           ;LOCK ON HARD ERROR
```

```
1714
1715  002630  005700                  A033:   TST     R0              ;"Z" SHOULD SET
1716  002632  001402                          BEQ     BT034           ;BR IF "Z" SET
1717
1718  002634  000000                  E2033:  HALT                    ;CMP FAILED TO AUTO INC. R0
1719  002636  000760                          BR      BT033           ;LOCK ON HARD ERROR
1720                                  ; ******************************************************************
1721                                          .SBTTL BT034 "CMP  (RA)+,RB" TEST WITH [SOURCE]<[RB]
1722                                  ; ******************************************************************
1723
1724  002640  012700  177776          BT034:  MOV     #PSW,R0         ;R0 POINTS TO PSW
1725  002644  012737  000330  177776          MOV     #330,@#PSW      ;MAKE [PSW]=330
1726  002652  012701  000340                  MOV     #340,R1         ;MAKE [DEST]=340
1727  002656  000264                          SEZ                     ;SET THE "Z" BIT
1728
1729  002660  022001                  1034:   CMP     (R0)+,R1        ;TEST THE CMP
1730
1731  002662  001002                          BNE     A034            ;BR IF "Z" GOT CLEARED
1732
1733  002664  000000                  EA034:  HALT                    ;CMP FAILED TO ACCESS PSW
1734  002666  000764                          BR      BT034           ;LOCK ON HARD ERROR
1735
1736  002670  005700                  A034:   TST     R0              ;"Z" SHOULD SET
1737  002672  001402                          BEQ     BT035           ;BR IF "Z" SET
1738
1739  002674  000000                  E2034:  HALT                    ;CMP FAILED TO AUTO INC. R0
1740  002676  000760                          BR      BT034           ;LOCK ON HARD ERROR
1741                                  ; ******************************************************************
1742                                          .SBTTL BT035 "CMP  RA,RB" TEST WITH [RA] = [RB]
1743                                  ; ******************************************************************
1744
1745  002700  012700  125252          BT035:  MOV     #125252,R0      ;MAKE [R0] = 125252
1746  002704  010001                          MOV     R0,R1           ;MAKE [R1] = 125252
1747  002706  000257                          CCC                     ;SCOPE SYNC
1748
1749  002710  020100                  1035:   CMP     R1,R0           ;TEST THE CMP
1750
1751  002712  001402                          BEQ     BT036           ;BR IF "Z" GOT SET
1752
1753  002714  000000                  E035:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1754  002716  000770                          BR      BT035           ;LOCK ON HARD ERROR
1755                                  ; ******************************************************************
1756                                          .SBTTL BT036 "CMP  RA,RB" TEST WITH [RA] < [RB]
1757                                  ; ******************************************************************
1758
1759  002720  012700  025252          BT036:  MOV     #25252,R0       ;MAKE [R0] = 25252
1760  002724  005001                          CLR     R1              ;MAKE [R1] = 000000
1761  002726  000264                          SEZ                     ;SCOPE SYNC - SET "Z"
1762
1763  002730  020100                  1036:   CMP     R1,R0           ;TEST THE CMP
1764
1765  002732  001002                          BNE     BT037           ;BR IF "Z" GOT  CLEARED
1766
1767  002734  000000                  E036:   HALT                    ;ERROR - CMP FAILED TO SET "Z"
1768  002736  000770                          BR      BT036           ;LOCK ON HARD ERROR
1769                                  ; ******************************************************************
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
1826
1827  003044  005000          BT042:  CLR     R0              ;MAKE [R0] = 000000
1828  003046  005001                  CLR     R1              ;MAKE [R1] = 000000
1829  003050  000257                  CCC                     ;SCOPE SYNC
1830
1831  003052  074100          I042:   XOR     R1,R0           ;TEST THE XOR
1832
1833  003054  005700                  TST     R0              ;RESULT = 000000 ?
1834  003056  001402                  BEQ     BT043           ;BR IF YES
1835
1836  003060  000000          E042:   HALT                    ;XOR FAILED
1837  003062  000770                  BR      BT042
1838
1839                          ; ******************************************************************
1840                                  .SBTTL BT043 "XOR  RA,RB" TEST WITH [RA] = [RB] = 177777
1841                          ; ******************************************************************
1842
1843  003064  005000          BT043:  CLR     R0              ;MAKE [R0] = 177777
1844  003066  005100                  COM     R0
1845  003070  010001                  MOV     R0,R1           ;MAKE [R1] = 177777
1846  003072  000257                  CCC                     ;SCOPE SYNC
1847
1848  003074  074100          I043:   XOR     R1,R0           ;TEST THE XOR
1849
1850  003076  005700                  TST     R0              ;RESULT = 000000 ?
1851  003100  001402                  BEQ     BT044   ;BR IF YES
1852
1853  003102  000000          E043:   HALT                    ;XOR FAILED
1854  003104  000767                  BR      BT043           ;LOCK ON HARD ERROR
1855
1856                          ; ******************************************************************
1857                                  .SBTTL BT044 "XOR  RA,RB" TEST WITH [RB]=052525,[RA]=125252
1858                          ; ******************************************************************
1859
1860  003106  012701  125252  BT044:  MOV     #125252,R1      ;MAKE [R1]=125252
1861  003112  012700  052525          MOV     #052525,R0      ;MAKE [R0]=052525
1862  003116  000257                  CCC                     ;SCOPE SYNC
1863
1864  003120  074100          I044:   XOR     R1,R0           ;TEST THE XOR
1865
1866  003122  020027  177777          CMP     R0,#-1          ;RESULT = 177777 ?
1867  003126  001402                  BEQ     BT045           ;BR IF YES
1868
1869  003130  000000          E044:   HALT                    ;XOR FAILED
1870  003132  000400                  BR      BT045           ;LOCK ON HARD ERROR
1871                          ; ******************************************************************
1872                                  .SBTTL BT045 "XOR  RA,RB" TEST WITH [RA]=052525,[RB]=125252
1873                          ; ******************************************************************
1874
1875  003134  012700  125252  BT045:  MOV     #125252,R0      ;MAKE [R0]=125252
1876  003140  012701  052525          MOV     #052525,R1      ;MAKE [R1]=052525
1877  003144  000257                  CCC                     ;SCOPE SYNC
1878
1879  003146  074100          I045:   XOR     R1,R0           ;TEST THE XOR
1880
1881  003150  020027  177777          CMP     R0,#-1          ;RESULT = 177777 ?
```

```
1882  003154  001402                  BEQ     BT046           ;BR IF YES
1883
1884  003156  000000          E045:   HALT                    ;XOR FAILED
1885  003160  000765                  BR      BT045           ;LOCK ON HARD ERROR
1886
1887                          ; ***************************************************************
1888                                  .SBTTL BT046 GPR ADDRESS INTERRACTION TEST
1889                          ; ***************************************************************
1890
1891  003162  012700  125252  BT046:  MOV     #125252,R0      ;[R0] = 125252
1892  003166  010001                  MOV     R0,R1
1893  003170  005101                  COM     R1              ;[R1] = 052525
1894  003172  010102                  MOV     R1,R2
1895  003174  005102                  COM     R2              ;[R2] = 125252
1896  003176  010203                  MOV     R2,R3
1897  003200  005103                  COM     R3              ;[R3] = 052525
1898  003202  010304                  MOV     R3,R4
1899  003204  005104                  COM     R4              ;[R4] = 125252
1900  003206  010405                  MOV     R4,R5
1901  003210  005105                  COM     R5              ;[R5] = 052525
1902
1903  003212  074100          I046:   XOR     R1,R0           ;[R0] S/B = 177777
1904  003214  074200                  XOR     R2,R0           ;[R0] S/B = 125252
1905  003216  074300                  XOR     R3,R0           ;[R0] S/B = 177777
1906  003220  074400                  XOR     R4,R0           ;[R0]  S/B = 125252
1907  003222  074500                  XOR     R5,R0           ;[R0]  S/B = 177777
1908  003224  005100                  COM     R0              ;[R0] S/B = 000000
1909
1910  003226  001402                  BEQ     A046            ;BR IF [R0] WAS 000000
1911
1912  003230  000000          EA046:  HALT                    ;GPR ADDRESSING PROBLEM
1913  003232  000753                  BR      BT046           ;LOCK ON HARD ERROR
1914
1915  003234  020627  001000  A046:   CMP     SP,#STACK       ;DID R6 GET DISTURBED
1916  003240  001402                  BEQ     BASIC           ;BR IF NOT
1917
1918  003242  000000          E2046:  HALT                    ;R6 ADDRESS PROBLEM
1919  003244  000746                  BR      BT046           ;LOCK ON HARD ERROR
```

```
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                ;          /  /  /  /  /  / BASIC INSTRUCTION TESTS /  /  /  /  /  /
                                ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

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

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

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

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

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

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

003342  000402                          BR      TST2            ;;GO TO NEXT TEST

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

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

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

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

```
                                ;*TEST 3        BASIC "BEQ" TEST WITH Z=0
                                ;;**************************************************************
                                TST3:
003364                                  MOV     #3,R0           ;;LOAD R0 WITH TEST NUMBER
003364  012700  000003          1$:     CCC                     ;MAKE Z=0
003370  000257
                                2$:     BEQ     3$              ;TEST THE BEQ - IT SHOULD NOT BR
003372  001401
                                        BR      TST4            ;;GO TO NEXT TEST
003374  000402
                                3$:     HALT                    ;BEQ BRANCHED WITH Z=0
003376  000000                          BR      1$              ;LOCK ON HARD ERROR
003400  000773

                                ;;**************************************************************
                                ;*TEST 4        BASIC "BPL" TEST WITH N=1
                                ;;**************************************************************
003402                          TST4:
003402  012700  000004                  MOV     #4,R0           ;;LOAD R0 WITH TEST NUMBER
003406  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003412  000270                          SEN                     ;MAKE N=1

003414  100001                2$:     BPL     3$              ;TEST THE BPL - IT SHOULDN'T BR

003416  000402                        BR      TST5            ;;GO TO NEXT TEST

003420  000000                3$:     HALT                    ;BPL BRANCHED WITH N=1
003422  000771                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 5        BASIC "BPL" TEST WITH N=0
                                ;;**************************************************************
003424                          TST5:
003424  012700  000005                  MOV     #5,R0           ;;LOAD R0 WITH TEST NUMBER
003430  005037  177776          1$:     CLR     @#PSW           ;CLEAR THE PSW
003434  000257                          CCC                     ;SCOPE SYNC

003436                          2$:
003436  100002                        BPL     TST6            ;;TEST THE BPL - IT SHOULD BR

003440  000000                5$:     HALT                    ;BPL FAILED TO LOAD THE PC
003442  000772                        BR      1$              ;LOCK ON HARD ERROR

                                ;;**************************************************************
                                ;*TEST 6        BASIC "MOV  (RA),RB" TEST - (RA)=177776
                                ;;**************************************************************
003444                          TST6:
003444  012700  000006                  MOV     #6,R0           ;;LOAD R0 WITH TEST NUMBER
003450  012705  177776                  MOV     #PSW,R5         ;SOURCE ADDR = 177776
003454  005015          1$:     CLR     (R5)            ;MAKE [PSW]=000
003456  005003                  CLR     R3              ;[DEST] = 000000
003460  000277                  SCC                     ;MAKE [PSW]=017

003462  011503                2$:     MOV     (R5),R3         ;TEST THE MOV

003464  020327  000017                CMP     R3,#17          ;CORRECT RESULT ?
003470  001402                        BEQ     TST7            ;;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 7         BASIC "CMP  RA,(RB)" TEST - [RA] = [DEST]
                                ;;*****************************************************************
003476                          TST7:
003476  012700  000007          3$:     MOV     #7,R0           ;;LOAD R0 WITH TEST NUMBER
003502  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
003506  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
003512  012737  125252  063312  1$:     MOV     #125252,@#MBUF0 ;MAKE [DEST] = 125252
003520  000257                          CCC                     ;MAKE N:C=0000
                                
003522  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
                                
003524  001402                          BEQ     TST10           ;;BR IF "Z" GOT SET
                                
003526  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO SET "Z"
003530  000770                          BR      1$              ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 10        BASIC "CMP  RA,(RB)" TEST - [RA] NOT EQUAL TO  [DEST]
                                ;;*****************************************************************
003532                          TST10:
003532  012700  000010                  MOV     #10,R0          ;;LOAD R0 WITH TEST NUMBER
003536  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
003542  012704  000001                  MOV     #1,R4           ;RESULT S / B = 000001
003546  005037  063312  1$:     CLR     @#MBUF0         ;MAKE [DEST] = 000000
003552  000264                          SEZ                     ;MAKE N:C=0100
                                
003554  020412                  2$:     CMP     R4,(R2)         ;TEST THE CMP
                                
003556  001002                          BNE     TST11           ;;BR IF "Z" GOT CLEARED
                                
003560  000000                  3$:     HALT                    ;ERROR - CMP FAILED TO CLR "Z"
003562  000771                          BR      1$              ;LOCK ON HARD ERROR
                                
                                ;;*****************************************************************
                                ;*TEST 11        BASIC "CMP #N,R" TEST - N = [R]
                                ;;*****************************************************************
003564                          TST11:
003564  012700  000011                  MOV     #11,R0          ;;LOAD R0 WITH TEST NUMBER
003570  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
003574  010403                  1$:     MOV     R4,R3           ;[DEST] = 125252
003576  000257                          CCC                     ;SCOPE SYNC
                                
003600  022703  125252          2$:     CMP     #125252,R3      ;TEST THE CMP
                                
003604  001402                          BEQ     4$              ;BR IF N = [R]
                                
003606  000000                  3$:     HALT                    ;CMP FAILED
003610  000771                          BR      1$              ;LOCK ON HARD ERROR
                                
003612  020403                  4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
003614  001402                          BEQ     TST12           ;;BR IF NO
                                
003616  000000                  5$:     HALT                    ;CMP DELIVERED A RESULT
```

```
2088  003620  000765                    BR      1$              ;LOCK ON HARD ERROR
2089
2090                                    ;;*******************************************************************
2091                                    ;*TEST 12       BASIC "CMP #N,R" TEST - N NOT EQUAL TO [R]
2092                                    ;;*******************************************************************
2093  003622                            TST12:
2094  003622  012700  000012                   MOV     #12,R0          ;;LOAD R0 WITH TEST NUMBER
2095  003626  005004                            CLR     R4              ;RESULT S / B = 000000
2096  003630  010403                    1$:     MOV     R4,R3           ;[DEST] = 125252
2097  003632  000264                            SEZ                     ;SCOPE SYNC
2098
2099  003634  022703  000001            2$:     CMP     #1,R3           ;TEST THE CMP
2100
2101  003640  001002                            BNE     4$              ;BR IF N NOT EQUAL TO [R]
2102
2103  003642  000000                    3$:     HALT                    ;CMP FAILED
2104  003644  000771                            BR      1$              ;LOCK ON HARD ERROR
2105
2106  003646  020403                    4$:     CMP     R4,R3           ;DID CMP ALTER [DEST]?
2107  003650  001402                            BEQ     TST13           ;;BR IF NO
2108
2109  003652  000000                    5$:     HALT                    ;CMP DELIVERED A RESULT
2110  003654  000765                            BR      1$              ;LOCK ON HARD ERROR
2111
2112                                    ;;*******************************************************************
2113                                    ;*TEST 13       BASIC "MOV  RA,(RB)" TEST
2114                                    ;;*******************************************************************
2115  003656                            TST13:
2116  003656  012700  000013                   MOV     #13,R0          ;;LOAD R0 WITH TEST NUMBER
2117  003662  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR=MBUF0
2118  003666  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
2119  003672  005012                    1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2120  003674  000257                            CCC                     ;SCOPE SYNC - N:C=0000
2121
2122  003676  010412                    2$:     MOV     R4,(R2)         ;TEST THE MOV
2123
2124  003700  020412                            CMP     R4,(R2)         ;RESULT CORRECT ?
2125  003702  001402                            BEQ     TST14           ;;BR IF  YES
2126
2127  003704  000000                    3$:     HALT                    ;ERROR - MOV FAILED
2128  003706  000771                            BR      1$              ;LOCK ON HARD ERROR
2129
2130                                    ;;*******************************************************************
2131                                    ;*TEST 14       BASIC "MOV  #N,(R)" TEST
2132                                    ;;*******************************************************************
2133  003710                            TST14:
2134  003710  012700  000014                   MOV     #14,R0          ;;LOAD R0 WITH TEST NUMBER
2135  003714  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2136  003720  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
2137  003724  005012                    1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2138  003726  000257                            CCC                     ;SCOPE SYNC
2139
2140  003730  012712  177777            2$:     MOV     #-1,(R2)        ;TEST THE MOV
2141
2142  003734  020412                            CMP     R4,(R2)         ;RESULT OK ?
2143  003736  001402                            BEQ     TST15           ;;BR IF YES
```

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

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

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

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

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

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

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

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

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

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

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

```
                                ;;*****************************************************************
                                ;*TEST 20       BASIC "TST  @#A" TEST WITH [A] LT 0
                                ;;*****************************************************************
004110                          TST20:
004110  012700  000020                  MOV     #20,R0          ;;LOAD R0 WITH TEST NUMBER
004114  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004120  012704  100000                  MOV     #100000,R4      ;MAKE S / B = 100000
004124  010412                  1$:     MOV     R4,(R2)         ;MAKE [DEST] = 100000
004126  000257                          CCC                     ;SCOPE SYNC

004130  005737  063312          2$:     TST     @#MBUF0         ;TEST THE TST

004134  001401                          BEQ     3$              ;BR IF "Z" SET - IT SHOULDN'T BE
004136  100402                          BMI     4$              ;BR IF "N" SET - IT SHOULD BE

004140  000000                  3$:     HALT                    ;TST FAILED TO ALTER CODES PROPERLY
004142  000770                          BR      1$              ;LOCK ON HARD ERROR
004144  020412                  4$:     CMP     R4,(R2)         ;DID TST DISTURB [DEST] ?
004146  001402                          BEQ     TST21           ;;BR IF NOT

004150  000000                  5$:     HALT                    ;TST DELIVERED A RESULT
004152  000764                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 21       BASIC "TST  @#A" WITH [A] = 0
                                ;;*****************************************************************
004154                          TST21:
004154  012700  000021                  MOV     #21,R0          ;;LOAD R0 WITH TEST NUMBER
004160  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004164  005004                          CLR     R4              ;RESULT S / B = 0 (IT SHOULDN'T CHANGE
004166  005012                  1$:     CLR     (R2)            ;[DEST] = 0
004170  000257                          CCC                     ;SCOPE SYNC - Z-0

004172  005737  063312          2$:     TST     @#MBUF0         ;TEST THE TST

004176  001402                          BEQ     4$              ;BR IF TST SET "Z"

004200  000000                  3$:     HALT                    ;TST FAILED TO SET "Z"
004202  000771                          BR      1$              ;LOCK ON HARD ERROR

004204  020412                  4$:     CMP     R4,(R2)         ;[DEST] STILL = 000000
004206  001402                          BEQ     TST22           ;;BR IF YES

004210  000000                  5$:     HALT                    ;TST ALTERED THE [DEST]
004212  000765                          BR      1$              ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 22       BASIC "BIT  #N,@#A" WITH BIT SET IN "A"
                                ;;*****************************************************************
004214                          TST22:
004214  012700  000022                  MOV     #22,R0          ;;LOAD R0 WITH TEST NUMBER
004220  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
004224  012704  040000                  MOV     #40000,R4       ;RESULT S / B = 40000
```

```
2256  004230  010412                  1$:     MOV     R4,(R2)         ;MAKE [DEST] = 40000
2257  004232  000277                          SCC                     ;SCOPE SYNC - Z=1
2258
2259  004234  032737  040000  063312  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
2260
2261  004242  001002                          BNE     TST23           ;;BR IF Z=0 - IT SHOULD BE
2262
2263  004244  000000                  3$:     HALT                    ;BIT FAILED TO CLEAR "Z"
2264  004246  000770                          BR      1$              ;LOCK ON HARD ERROR
2265
2266                                  ;;**********************************************************
2267                                  ;*TEST 23        BASIC "BIT  #N,@#A" WITH BIT CLEAR IN "A"
2268                                  ;;**********************************************************
2269  004250                          TST23:
2270  004250  012700  000023                  MOV     #23,R0          ;;LOAD R0 WITH TEST NUMBER
2271  004254  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2272  004260  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2273  004262  000257                          CCC                     ;SCOPE SYNC - Z=0
2274
2275  004264  032737  040000  063312  2$:     BIT     #40000,@#MBUF0  ;TEST THE BIT
2276
2277  004272  001402                          BEQ     4$              ;BR IF Z=1 - IT SHOULD BE
2278
2279  004274  000000                  3$:     HALT                    ;BIT FAILED TO SET "Z"
2280  004276  000770                          BR      1$              ;LOCK ON HARD ERROR
2281
2282  004300  005712                  4$:     TST     (R2)            ;DID BIT DELIVER A RESULT
2283  004302  001402                          BEQ     TST24           ;;BR IF NOT
2284
2285  004304  000000                  5$:     HALT                    ;BIT DISTURBED THE [DEST]
2286  004306  000764                          BR      1$              ;LOCK ON HARD ERROR
2287
2288                                  ;;**********************************************************
2289                                  ;*TEST 24        BASIC "TST  (R)+" TEST
2290                                  ;;**********************************************************
2291  004310                          TST24:
2292  004310  012700  000024                  MOV     #24,R0          ;;LOAD R0 WITH TEST NUMBER
2293                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT0
2294  004314  032737  000001  063234          BIT     #BIT0,@#BPTLOC  ;BREAKPOINT HALT SET ??
2295  004322  001401                          BEQ     .+4             ;BR IF NOT
2296  004324  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2297  004326  012702  063312                  MOV     #MBUF0,R2       ;INITIAL DEST ADDR = MBUF0
2298  004332  005012                  1$:     CLR     (R2)            ;MAKE [DEST] = 000000
2299  004334  000257                          CCC                     ;SCOPE SYNC
2300
2301  004336  005722                  2$:     TST     (R2)+           ;TEST THE TST
2302
2303  004340  001402                          BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2304
2305  004342  000000                  3$:     HALT                    ;TST FAILED TO SET "Z"
2306  004344  000772                          BR      1$              ;LOCK ON HARD ERROR
2307
2308  004346  022702  063314          4$:     CMP     #MBUF0+2,R2     ;DID REG. GET AUTO-INCREMENTED ?
2309  004352  001402                          BEQ     TST25           ;;BR IF YES
2310
2311  004354  000000                  5$:     HALT                    ;TST FAILED TO UPDATE REGISTER
```

```
                                                H 4
  2312  004356  000765                  BR      1$              ;LOCK ON HARD ERROR
  2313
  2314                                  ;;*****************************************************************
  2315                                  ;*TEST 25       BASIC "TST -(R)" TEST
  2316                                  ;;*****************************************************************
  2317  004360                          TST25:
  2318  004360  012700  000025          MOV     #25,R0          ;;LOAD R0 WITH TEST NUMBER
  2319  004364  012702  063330          MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
  2320  004370  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
  2321  004374  012705  063332  1$:     MOV     #DWTA+10,R5     ;BASE DEST ADDR = DWTA+10
  2322  004400  000270                  SEN                     ;SCOPE SYNC
  2323
  2324  004402  005745          2$:     TST     -(R5)           ;TEST THE TST
  2325
  2326  004404  100002                  BPL     4$              ;BR IF "N" CLEAR
  2327
  2328  004406  000000          3$:     HALT                    ;TST FAILED TO CLEAR "N"
  2329  004410  000771                  BR      1$              ;LOCK ON HARD ERROR
  2330
  2331  004412  020502          4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED?
  2332  004414  001402                  BEQ     6$              ;BR IF YES
  2333
  2334  004416  000000          5$:     HALT                    ;ERROR - TST FAILED TO UPDATE DEST REG
  2335  004420  000765                  BR      1$              ;LOCK ON HARD ERROR
  2336
  2337  004422  020412          6$:     CMP     R4,(R2)         ;DID TST ALTER [DEST]?
  2338  004424  001403                  BEQ     TST26           ;;BR IF NOT
  2339
  2340  004426  000000          7$:     HALT                    ;TST ALTERED [DEST]
  2341  004430  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
  2342  004432  000760                  BR      1$              ;LOCK ON HARD ERROR
  2343
  2344                                  ;;*****************************************************************
  2345                                  ;*TEST 26       BASIC "COM  @#A" TEST
  2346                                  ;;*****************************************************************
  2347  004434                          TST26:
  2348  004434  012700  000026          MOV     #26,R0          ;;LOAD R0 WITH TEST NUMBER
  2349  004440  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
  2350  004444  005004                  CLR     R4              ;RESULT S / B = 177777
  2351  004446  005104                  COM     R4
  2352  004450  005012          1$:     CLR     (R2)            ;MAKE [DEST] = 000000
  2353  004452  000257                  CCC                     ;SCOPE SYNC
  2354
  2355  004454  005137  063312  2$:     COM     @#MBUF0 ;TEST THE COM
  2356
  2357  004460  020412                  CMP     R4,(R2)         ;RESULT = 177777 ??
  2358  004462  001402                  BEQ     TST27           ;;BR IF YES
  2359
  2360  004464  000000          3$:     HALT                    ;COM DELIVERED THE WRONG RESULT
  2361  004466  000770                  BR      1$
  2362
  2363                                  ;;*****************************************************************
  2364                                  ;*TEST 27       BASIC "INC  @#A" TEST
  2365                                  ;;*****************************************************************
  2366  004470                          TST27:
  2367  004470  012700  000027          MOV     #27,R0          ;;LOAD R0 WITH TEST NUMBER
```

```
2368  004474  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2369  004500  012704  000100          MOV     #100,R4         ;RESULT S / B = 100
2370  004504  012712  000077  1$:     MOV     #77,(R2)        ;[DEST] = 77
2371  004510  000257                  CCC                     ;SCOPE SYNC
2372
2373  004512  005237  063312  2$:     INC     @#MBUF0         ;TEST THE INC
2374
2375  004516  020412                  CMP     R4,(R2)         ;DID RESULT = 100 ??
2376  004520  001402                  BEQ     TST30           ;;BR IF YES
2377
2378  004522  000000  3$:     HALT                            ;INC DELIVERED WRONG RESULT
2379  004524  000767                  BR      1$              ;LOCK ON HARD ERROR
2380
2381                          ;;*****************************************************************
2382                          ;*TEST 30        BASIC "DEC  RN" TEST
2383                          ;;*****************************************************************
2384  004526                  TST30:
2385  004526  012700  000030          MOV     #30,R0          ;;LOAD R0 WITH TEST NUMBER
2386  004532  012703  000001  1$:     MOV     #1,R3           ;[DEST] = +1
2387  004536  000257                  CCC                     ;SCOPE SYNC
2388
2389  004540  005303  2$:     DEC     R3                      ;TEST THE DEC
2390
2391  004542  005703                  TST     R3              ;RESULT = 000000 ??
2392  004544  001402                  BEQ     TST31           ;;BR IF YES
2393
2394  004546  000000  3$:     HALT                            ;DEC DELIVERED THE WRONG RESULT
2395  004550  000770                  BR      1$              ;LOCK ON HARD ERROR
2396
2397                          ;;*****************************************************************
2398                          ;*TEST 31        BASIC "DEC  @#A" TEST
2399                          ;;*****************************************************************
2400  004552                  TST31:
2401  004552  012700  000031          MOV     #31,R0          ;;LOAD R0 WITH TEST NUMBER
2402  004556  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
2403  004562  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2404  004566  005012  1$:     CLR     (R2)                    ;MAKE [DEST] = 000000
2405  004570  000257                  CCC                     ;SCOPE SYNC
2406
2407  004572  005337  063312  2$:     DEC     @#MBUF0         ;TEST THE DEC
2408
2409  004576  020412                  CMP     R4,(R2)         ;DID RESULT = 177777 ??
2410  004600  001402                  BEQ     TST32           ;;BR IF YES
2411
2412  004602  000000  3$:     HALT                            ;DEC DELIVERED WRONG RESULT
2413  004604  000770                  BR      1$              ;LOCK ON HARD ERROR
2414
2415                          ;;*****************************************************************
2416                          ;*TEST 32        BASIC "CLR X(R)" TESTS
2417                          ;;*****************************************************************
2418  004606                  TST32:
2419  004606  012700  000032          MOV     #32,R0          ;;LOAD R0 WITH TEST NUMBER
2420  004612  012702  063314          MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
2421  004616  005004                  CLR     R4              ;RESULT S / B = 000000
2422  004620  012705  063312  1$:     MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
2423  004624  012712  177777          MOV     #-1,(R2)        ;[DEST]  177777
```

```
 2424  004630  000257                    CCC                     ;SCOPE SYNC
 2425
 2426  004632  005065  000002  2$:       CLR     2(R5)           ;TEST THE CLR
 2427
 2428  004636  020412                    CMP     R4,(R2)         ;RESULT = 0?
 2429  004640  001402                    BEQ     TST33           ;;BR IF YES
 2430
```

```
2431  004642  000000                  3$:     HALT                    ;CLR FAILED TO ZERO [DEST]
2432  004644  000765                          BR      1$              ;LOCK ON HARD ERROR.
2433
2434                                  ;;*******************************************************************
2435                                  ;*TEST 33       BASIC "ASL  RN" TEST WITH [DEST]=125252 AND C(0)
2436                                  ;;*******************************************************************
2437  004646                          TST33:
2438  004646  012700  000033                MOV     #33,R0          ;;LOAD R0 WITH TEST NUMBER
2439  004652  012703  125252          1$:   MOV     #125252,R3      ;MAKE [DEST] = 125252
2440  004656  000257                        CCC                     ;MAKE C=0
2441
2442  004660  006303                  2$:   ASL     R3              ;TEST THE ASL - IT SHOULD SET "C"
2443
2444  004662  103402                        BCS     4$              ;BR IF "C" GOT SET
2445
2446  004664  000000                  3$:   HALT                    ;ASL FAILED TO SET "C" BIT
2447  004666  000771                        BR      1$              ;LOCK ON HRD ERROR
2448
2449  004670  022703  052524          4$:   CMP     #52524,R3       ;WAS RESULT = 52524 ??
2450  004674  001402                        BEQ     TST34           ;;BR IF YES
2451
2452  004676  000000                  5$:   HALT                    ;ASL DELIVERED THE WRONG RESULT
2453  004700  000764                        BR      1$              ;LOCK ON HARD ERROR
2454
2455                                  ;;*******************************************************************
2456                                  ;*TEST 34       BASIC "ASL  RN" TEST WITH [DEST]=052525 AND C(1)
2457                                  ;;*******************************************************************
2458  004702                          TST34:
2459  004702  012700  000034                MOV     #34,R0          ;;LOAD R0 WITH TEST NUMBER
2460  004706  012703  052525          1$:   MOV     #052525,R3      ;MAKE [DEST] = 052525
2461  004712  000261                        SEC                     ;MAKE C=1
2462
2463  004714  006303                  2$:   ASL     R3              ;TEST THE ASL - IT SHOULD CLR "C"
2464
2465  004716  103002                        BCC     4$              ;BR IF "C" GOT CLEARED
2466
2467  004720  000000                  3$:   HALT                    ;ASL FAILED TO CLEAR "C"
2468  004722  000771                        BR      1$              ;LOCK ON HARD ERROR
2469
2470  004724  022703  125252          4$:   CMP     #125252,R3      ;RESULT = 125252 ??
2471  004730  001402                        BEQ     TST35           ;,BR IF YES
2472
2473  004732  000000                  5$:   HALT                    ;ASL DELIVERED WRONG REUL'
2474  004734  000764                        BR      1$              ;LOCK ON HARD ERROR
2475                                  ;;*******************************************************************
2476                                  ;*TEST 35       BASIC "ROL  RN" TEST WITH [DEST]=125252 AND C(0)
2477                                  ;;*******************************************************************
2478  004736                          TST35:
2479  004736  012700  000035                MOV     #35,R0          ;;LOAD R0 WITH TEST NUMBER
2480  004742  012703  125252          1$:   MOV     #125252,R3      ;MAKE [DEST] = 125252
2481  004746  000257                        CCC                     ;MAKE C=0
2482
2483  004750  006103                  2$:   ROL     R3              ;TEST THE ROL - IT SHOULD SET C
2484
2485  004752  103402                        BCS     4$              ;BR IF "C" GOT SET
2486
```

```
2487  004754  000000                3$:     HALT                    ;ROL FAILED TO SET "C"
2488  004756  000771                        BR      1$              ;LOCK ON HARD ERROR
2489
2490  004760  022703  052524        4$:     CMP     #052524,R3      ;RESULT = 052524 ??
2491  004764  001402                        BEQ     TST36           ;;BR IF YES
2492
2493  004766  000000                5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2494  004770  000764                        BR      1$              ;LOCK ON HARD ERROR
2495
2496                                ;;**********************************************************************
2497                                ;*TEST 36       BASIC "ROL  RN" TEST WITH [DEST]=052524 AND C(1)
2498                                ;;**********************************************************************
2499  004772                        TST36:
2500  004772  012700  000036                MOV     #36,R0          ;;LOAD R0 WITH TEST NUMBER
2501  004776  012703  052524        1$:     MOV     #052524,R3      ;MAKE [DEST] = 052524
2502  005002  000261                        SEC                     ;MAKE C=1
2503
2504  005004  006103                2$:     ROL     R3              ;TEST THE ROL - IT SHOULD CLEAR C
2505
2506  005006  103002                        BCC     4$              ;BR IF "C" IS CLEAR
2507
2508  005010  000000                3$:     HALT                    ;ROL FAILED TO CLEAR "C"
2509  005012  000771                        BR      1$              ;LOCK ON HARD ERROR
2510
2511  005014  022703  125251        4$:     CMP     #125251,R3      ;RESULT = 125251 ??
2512  005020  001402                        BEQ     TST37           ;;BR IF YES
2513
2514  005022  000000                5$:     HALT                    ;ROL DELIVERED WRONG RESULT
2515  005024  000764                        BR      1$              ;LOCK ON HARD ERROR
2516
2517                                ;;**********************************************************************
2518                                ;*TEST 37       BASIS "TSTB (R)" TEST - EVEN ADDRESS
2519                                ;;**********************************************************************
2520  005026                        TST37:
2521  005026  012700  000037                MOV     #37,R0          ;;LOAD R0 WITH TEST NUMBER
2522  005032  012702  063330                MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2523  005036  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
2524  005042  000257                1$:     CCC                     ;SCOPE SYNC
2525
2526  005044  105712                2$:     TSTB    (R2)            ;TEST THE TSTB
2527
2528  005046  100402                        BMI     4$              ;BR IF "N" SET - IT SHOULD BE
2529
2530  005050  000000                3$:     HALT                    ;TSTB FAILED TO SET "N"
2531  005052  000773                        BR      1$              ;LOCK ON HARD ERROR
2532
2533  005054  020412                4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2534  005056  001403                        BEQ     TST40           ;;BR IF NOT
2535
2536  005060  000000                5$:     HALT                    ;TSTB ALTERED [DEST]
2537  005062  010412                        MOV     R4,(R2)         ;RESTORE [DEST]
2538  005064  000766                        BR      1$              ;LOCK ON HARD EROR
2539                                ;;**********************************************************************
2540                                ;*TEST 40       BASIS "TSTB (R)" TEST - ODD ADDRESS
2541                                ;;**********************************************************************
2542  005066                        TST40:
```

```
2543  005066  012700  000040          MOV     #40,R0          ;;LOAD R0 WITH TEST NUMBER
2544  005072  012702  064040          MOV     #DWTB+6,R2      ;DEST ADDR = DWTB+6
2545  005076  012704  177401          MOV     #177401,R4      ;RESULT S / B = 177401
2546  005102  012703  064041          MOV     #DWTB+7,R3      ;DEST ADDR USED = DWTB+7
2547  005106  000257          1$:     CCC                     ;SCOPE SYNC
2548
2549  005110  105713          2$:     TSTB    (R3)            ;TEST THE TSTB
2550
2551  005112  100402                  BMI     4$              ;BR IF "N" SET - IT SHOULD BE
2552
2553  005114  000000          3$:     HALT                    ;TSTB FAILED TO SET "N"
2554  005116  000773                  BR      1$              ;LOCK ON HARD ERROR
2555
2556  005120  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]
2557  005122  001403                  BEQ     TST41           ;;BR IF NOT
2558
2559  005124  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2560  005126  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2561  005130  000766                  BR      1$              ;LOCK ON HARD EROR
2562
2563                          ;;**********************************************************************
2564                          ;*TEST 41        BASIC "TSTB @#A" TEST - EVEN ADDRESS
2565                          ;;**********************************************************************
2566  005132                  TST41:
2567  005132  012700  000041          MOV     #41,R0          ;;LOAD R0 WITH TEST NUMBER
2568  005136  012702  063326          MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
2569  005142  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
2570  005146  000257          1$:     CCC                     ;SCOPE SYNC
2571
2572  005150  105737  063326  2$:     TSTB    @#DWTA+4        ;TEST THE TSTB
2573
2574  005154  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2575
2576  005156  000000          3$:     HALT                    ;TSTB FAILED TO SET "Z"
2577  005160  000772                  BR      1$              ;LOCK ON HARD ERROR
2578
2579  005162  020412          4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
2580  005164  001403                  BEQ     TST42           ;;BR IF NOT
2581
2582  005166  000000          5$:     HALT                    ;TSTB ALTERED [DEST]
2583  005170  010412                  MOV     R4,(R2)         ;RESTORE [DEST]
2584  005172  000765                  BR      1$              ;LOCK ON HARD ERROR
2585
2586                          ;;**********************************************************************
2587                          ;*TEST 42        BASIC "TSTB @#A" TEST - ODD ADDRESS
2588                          ;;**********************************************************************
2589  005174                  TST42:
2590  005174  012700  000042          MOV     #42,R0          ;;LOAD R0 WITH TEST NUMBER
2591  005200  012702  063330          MOV     #DWTA+6,R2      ;DEST ADDR = DWTA+6
2592  005204  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
2593  005210  000257          1$:     CCC                     ;SCOPE SYNC
2594
2595  005212  105737  063331  2$:     TSTB    @#DWTA+7        ;TEST THE TSTB
2596
2597  005216  001402                  BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
2598
```

```
2599  005220  000000            3$:     HALT                    ;TSTB FAILED TO SET "Z"
2600  005222  000772                    BR      1$              ;LOCK ON HARD ERROR
2601
2602  005224  020412            4$:     CMP     R4,(R2)         ;DID TSTB DISTURB [DEST]?
2603  005226  001403                    BEQ     TST43           ;;BR IF NOT
2604
2605  005230  000000            5$:     HALT                    ;TSTB ALTERED [DEST]
2606  005232  010412                    MOV     R4,(R2)         ;RESTORE [DEST]
2607  005234  000765                    BR      1$              ;LOCK ON HARD ERROR
2608
2609                            ;;*******************************************************************
2610                            ;*TEST 43        BASIC "DECB 1(SP)"
2611                            ;;*******************************************************************
2612  005236                    TST43:
2613  005236  012700  000043            MOV     #43,R0          ;;LOAD R0 WITH TEST NUMBER
2614  005242  010605                    MOV     SP,R5           ;SAVE SP
2615  005244  012704  177400            MOV     #177400,R4      ;RESULT S / B = 177400
2616  005250  010506            1$:     MOV     R5,SP
2617  005252  005046                    CLR     -(SP)           ;[DEST] = 000000
2618  005254  000257                    CCC                     ;SCOPE SYNC
2619
2620  005256  105366  000001    2$:     DECB    1(SP)           ;TEST THE DECB
2621
2622  005262  020416                    CMP     R4,(SP)         ;RESULT = 177400?
2623  005264  001402                    BEQ     4$              ;BR IF YES
2624
2625  005266  000000            3$:     HALT                    ;ERROR - DFCB FAILED
2626  005270  000767                    BR      1$              ;LOCK ON HARD ERROR
2627
2628  005272  010506            4$:     MOV     R5,SP           ;RESET THE SP
2629
2630                            ;;*******************************************************************
2631                            ;*TEST 44        BASIC "MOV @#A,R" TEST
2632                            ;;*******************************************************************
2633  005274                    TST44:
2634  005274  012700  000044            MOV     #44,R0          ;;LOAD R0 WITH TEST NUMBER
2635  005300  005003            1$:     CLR     R3              ;[DEST] = 000000
2636  005302  000257                    CCC                     ;SCOPE SYNC
2637
2638  005304  013703  063276    2$:     MOV     @#ATA,R3        ;TEST THE MOV
2639
2640  005310  022703  063322            CMP     #DWTA,R3        ;RESULT = DWTA?
2641  005314  001402                    BEQ     TST45           ;;BR IF YES
2642
2643  005316  000000            3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2644  005320  000767                    BR      1$              ;LOCK ON HARD ERROR
2645
2646                            ;;*******************************************************************
2647                            ;*TEST 45        BASIC "MOV #N,X(R)" TEST
2648                            ;;*******************************************************************
2649  005322                    TST45:
2650  005322  012700  000045            MOV     #45,R0          ;;LOAD R0 WITH TEST NUMBER
2651  005326  012702  063314            MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0+2
2652  005332  012704  125252            MOV     #125252,R4      ;RESULT S / B = 125252
2653  005336  012703  063312    1$:     MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
2654  005342  005012                    CLR     (R2)            ;[DEST] = 000000
```

```
                                                ;SCOPE SYNC
                                                ;TEST THE MOV
                                                ;RESULT OK?
                                                ;;BR IF YES
                                                ;MOV DELIVERED WRONG RESULT
                                                ;LOCK ON HARD ERROR
                                ;;*****************************************************************
                                ;*TEST 46        BASIC "MOV #N,(R)" TEST
                                ;;*****************************************************************
                                TST46:
                                                ;;LOAD R0 WITH TEST NUMBER
                                                ;DEST ADDR = MBUF0
                                                ;RESULT S / B = 125252
                                1$:             ;[DEST] = 000000
                                                ;SCOPE SYNC

                                2$:             ;TEST THE MOV

                                                ;RESULT OK?
                                                ;;BR IF YES

                                3$:             ;MOV DELIVERED WRONG RESULT
                                                ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 47        BASIC "MOV (RA)+,RB" TEST
                                ;;*****************************************************************
                                TST47:
                                                ;;LOAD R0 WITH TEST NUMBER
                                1$:             ;SRC ADDR = ATA
                                                ;[DEST] = 000000
                                                ;SCOPE SYNC

                                2$:             ;TEST THE MOV

                                                ;RESULT OK?
                                                ;BR IF YES

                                3$:             ;MOV DELIVERED WRONG RESULT
                                                ;LOCK ON HARD ERROR

                                4$:             ;DID SRC REG GET INCREMENTED?
                                                ;;BR IF YES

                                5$:             ;MOV FAILED TO UPDATE SRC. REG.
                                                ;LOCK ON HARD ERROR

                                ;;*****************************************************************
                                ;*TEST 50        BASIC "MOV  @#A,@#B"
                                ;;*****************************************************************
                                TST50:
                                                ;;LOAD R0 WITH TEST NUMBER
```

```
 2711  005466  012702  063316            MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 2712  005472  012704  063322            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 2713  005476  005012                1$:  CLR     (R2)            ;MAKE [DEST] = 000000
 2714  005500  000257                     CCC                     ;SCOPE SYNC
 2715
 2716  005502  013737  063276  063316 2$:  MOV     @#ATA,@#MBUF1   ;TEST THE MOV
 2717  005510  020412                     CMP     R4,(R2)         ;DID RESULT = #DWTA ?
 2718  005512  001402                     BEQ     TST51           ;;BR IF YES
 2719
 2720  005514  000000                3$:  HALT                    ;MOV DELIVERED THE WRONG RESULT
 2721  005516  000767                     BR      1$              ;LOCK ON HARD ERROR
 2722
 2723                                ;;*****************************************************************
 2724                                ;*TEST 51        BASIC "MOV   X(R),PC" TEST
 2725                                ;;*****************************************************************
 2726  005520                        TST51:
 2727  005520  012700  000051            MOV     #51,R0          ;;LOAD R0 WITH TEST NUMBER
 2728  005524  012705  005532        1$: MOV     #2$,R5          ;[R5] = 2$ (BASE ADDRESS)
 2729  005530  000257                     CCC                     ;SCOPE SYNC
 2730
 2731  005532  016507  000010        2$: MOV     4$-2$(R5),PC    ;TEST THE MOV - GO TO NEXT TEST VIA 4$
 2732
 2733  005536  000000                3$:  HALT                    ;MOV FAILED TO LOAD THE PC
 2734  005540  000771                     BR      1$              ;LOCK ON HARD ERROR
 2735
 2736  005542  005544                4$:  .+2                     ;POINTER TO NEXT TEST
 2737
 2738                                ;;*****************************************************************
 2739                                ;*TEST 52        BASIC "MOV  @#A,(R)" TEST
 2740                                ;;*****************************************************************
 2741  005544                        TST52:
 2742  005544  012700  000052            MOV     #52,R0          ;;LOAD R0 WITH TEST NUMBER
 2743  005550  012704  063322            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
 2744  005554  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 2745  005560  005012                1$:  CLR     (R2)            ;MAKE [DEST]=000000
 2746  005562  000257                     CCC                     ;SCOPE SYNC - Z=0
 2747
 2748  005564  013712  063276        2$:  MOV     @#ATA,(R2)      ;TEST THE MOV
 2749
 2750  005570  020412                     CMP     R4,(R2)         ;DID RESULT = #DWTA ??
 2751  005572  001402                     BEQ     TST53           ;;BR IF YES
 2752
 2753  005574  000000                3$:  HALT                    ;MOV DELIVERED WRONG RESULT
 2754  005576  000770                     BR      1$              ;LOCK ON HARD ERROR
 2755
 2756                                ;;*****************************************************************
 2757                                ;*TEST 53        BASIC "MOV  X(RA),RB" TEST
 2758                                ;;*****************************************************************
 2759  005600                        TST53:
 2760  005600  012700  000053            MOV     #53,R0          ;;LOAD R0 WITH TEST NUMBER
 2761  005604  012705  063276            MOV     #ATA,R5         ;[R5] = BASE ADDR FOR SOURCE (ATA)
 2762  005610  005003                1$:  CLR     R3              ;MAKE [DEST] = 000000
 2763  005612  000257                     CCC                     ;SCOPE SYNC
 2764
 2765  005614  016503  000004        2$:  MOV     4(R5),R3        ;TEST THE MOV
 2766
```

```
2767  005620  022703  064630          CMP     #DBTA,P3        ;RESULT = #DBTA ??
2768  005624  001402                  BEQ     TST54           ;;BR IF YES
2769
2770  005626  000000          3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2771  005630  000767                  BR      1$              ;LOCK ON HARD ERROR
2772
2773                          ;;**********************************************************************
2774                          ;*TEST 54        BASIC "MOV  RA,-(RB)" TEST
2775                          ;;**********************************************************************
2776  005632                  TST54:
2777  005632  012700  000054          MOV     #54,R0          ;;LOAD R0 WITH TEST NUMBER
2778  005636  012702  063312          MOV     #MBUF0,R2       ;FINAL DEST ADDR = MBUF0
2779  005642  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
2780  005646  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = TEMP2 + 2
2781  005652  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
2782  005654  000257                  CCC                     ;SCOPE SYNC
2783
2784  005656  010445          2$:     MOV     R4,-(R5)        ;TEST THE MOV
2785
2786  005660  020412                  CMP     R4,(R2)         ;RESULT = 125252
2787  005662  001402                  BEQ     4$              ;BR IF YES
2788
2789  005664  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG ESULT
2790  005666  000767                  BR      1$              ;LOCK ON HARD ERROR
2791
2792  005670  020205          4$:     CMP     R2,R5           ;DID REGISTER GET DECREMENTED ?
2793  005672  001402                  BEQ     TST55           ;;BR IF YES
2794
2795  005674  000000          5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2796  005676  000763                  BR      1$              ;LOCK ON HARD ERROR
2797
2798                          ;;**********************************************************************
2799                          ;*TEST 55        BASIC "MOV   @#A,-(R)" TEST
2800                          ;;**********************************************************************
2801  005700                  TST55:
2802  005700  012700  000055          MOV     #55,R0          ;;LOAD R0 WITH TEST NUMBER
2803  005704  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
2804  005710  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2805  005714  012705  063314  1$:     MOV     #MBUF0+2,R5     ;INITIAL DEST ADDR = MBUF0+2
2806  005720  005012                  C       (R2)            ;MAKE [DEST] = 000000
2807  005722  000257                  CCC                     ;SCOPE SYNC
2808
2809  005724  013745  063276  2$:     MOV     @#ATA,-(R5)     ;TEST THE MOV
2810
2811  005730  020412                  CMP     R4,(R2)         ;RESULT = 000000
2812  005732  001402                  BEQ     4$              ;BR IF YES
2813
2814  005734  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2815  005736  000766                  BR      1$              ;LOCK ON HARD ERROR
2816
2817  005740  020502          4$:     CMP     R5,R2           ;DID DEST REG GET DECREMENTED ??
2818  005742  001402                  BEQ     TST56           ;;BR IF YES
2819
2820  005744  000000          5$:     HALT                    ,MOV FAILED TO UPDATE REGISTER
2821  005746  000762                  BR      1$              ;LOCK ON HARD ERROR
2822
```

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

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

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

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

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

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

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

006044  000000          3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
006046  000766                  BR      1$              ;LOCK ON HARD ERROR

006050  022705  063276  4$:     CMP     #ATA,R5         ;DID THE SRC REG GET DECREMENTED ?
006054  001402                  BEQ     TST60           ;;BR IF YES

006056  000000          5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REG
006060  000761                  BR      1$              ;LOCK ON HARD ERROR
                        ;;*****************************************************************
                        ;*TEST 60        BASIC "MOV   (RA),RB" TEST
                        ;;*****************************************************************
006062                  TST60:
006062  012700  000060          MOV     #60,R0          ;;LOAD R0 WITH TEST NUMBER
006066  012705  063276  1$:     MOV     #ATA,R5         ;INITIAL SOURCE ADDR = ATA
006072  005003                  CLR     R3              ;MAKE [DEST] - 000000
006074  000257                  CCC                     ;SCOPE SYNC

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

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

```
2879
2880  006106  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
2881  006110  000766                          BR      1$              ;LOCK ON HARD ERROR
2882
2883  006112  022705  063300          4$:     CMP     #A`A+2,R5       ;DID SOURCE REG GET INCREMENTED
2884  006116  001402                          BEQ     TST61           ;;BR IF YES
2885
2886  006120  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE SOURCE REGISTER
2887  006122  000761                          BR      1$              ;LOCK ON HARD ERROR
2888
2889                                  ;;**********************************************************************
2890                                  ;*TEST 61        BASIC "MOV   X(RA),RB" TEST
2891                                  ;;**********************************************************************
2892  006124                          TST61:
2893  006124  012700  000061                  MOV     #61,R0          ;;LOAD R0 WITH TEST NUMBER
2894  006130  012705  063276                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
2895  006134  005003                  1$:     CLR     R3              ;MAKE [DEST] = 000000
2896  006136  000257                          CCC                     ;SCOPE SYNC
2897
2898  006140  016503  000002          2$:     MOV     2(R5),R3        ;TEST THE MOV
2899
2900  006144  022703  064032                  CMP     #DWTB,R3        ;RESULT = #DWTB ?
2901  006150  001402                          BEQ     TST62           ;;BR IF YES
2902
2903  006152  000000                  3$:     HALT                    ;MOV FAILED TO DELIVER CORRECT RESULT
2904  006154  000767                          BR      1$              ;LOCK ON HARD ERROR
2905                                  ;;**********************************************************************
2906                                  ;*TEST 62        BASIC "MOV   @X(RA),RB" TEST
2907                                  ;;**********************************************************************
2908  006156                          TST62:
2909  006156  012700  000062                  MOV     #62,R0          ;;LOAD R0 WITH TEST NUMBER
2910  006162  012737  063324  063314          MOV     #DWTA+2,@#MBUF0+2 ;SET UP ADDRESS TABLE MBUF0
2911  006170  012705  063312                  MOV     #MBUF0,R5       ;BASE ADDRESS IN R5
2912  006174  005003                  1$:     CLR     R3              ;MAKE [DEST] = 000000
2913  006176  000257                          CCC                     ;SCOPE SYNC
2914
2915  006200  017503  000002          2$:     MOV     @2(R5),R3       ;TEST THE MOV
2916
2917  006204  022703  177777                  CMP     #-1,R3          ;RESULT = 177777
2918  006210  001402                          BEQ     TST63           ;;BR IF YES
2919
2920  006212  000000                  3$:     HALT                    ;MOV DELIVERED THE WRONG RESULT
2921  006214  000767                          BR      1$              ;LOCK ON HARD ERROR
2922
2923                                  ;;**********************************************************************
2924                                  ;*TEST 63        BASIC "MOV  (R)+,X(R)" TEST
2925                                  ;;**********************************************************************
2926  006216                          TST63:
2927  006216  012700  000063                  MOV     #63,R0          ;;LOAD R0 WITH TEST NUMBER
2928  006222  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
2929  006226  012702  063320                  MOV     #MBUF1+2,R2     ;FINAL DEST ADDR = MBUF1+2
2930  006232  010437  063312                  MOV     R4,@#MBUF0      ;SOURCE OPERAND = 125252
2931  006236  012705  063312          1$:     MOV     #MBUF0,R5       ;[R5] = INITIAL SRC ADDR - MBUF0
2932  006242  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
2933  006244  000257                          CCC                     ;SCOPE SYNC
2934
```

```
2935  006246  012565  000004          2$:     MOV     (R5)+,4(R5)     ;TEST THE MOV
2936
2937  006252  020412                          CMP     R4,(R2)         ;RESULT = 125252 ?
2938  006254  001402                          BEQ     4$              ;BR IF YES
2939
2940  006256  000000                  3$:     HALT                    ;MOV    DELIVERED WRONG RESULT
2941  006260  000766                          BR      1$              ;LOCK ON HARD ERROR
2942
2943  006262  022705  063314          4$:     CMP     #MBUF0+2,R5     ;DID REGISTER GET INCREMENTED ?
2944  006266  001402                          BEQ     TST64           ;;BR IF YES
2945
2946  006270  000000                  5$:     HALT                    ;MOV FAILED TO UPDATE REGISTER
2947  006272  000761                          BR      1$              ;LOCK ON HARD ERROR
2948
2949                                  ;;*****************************************************************
2950                                  ;*TEST 64        BASIC "CMP  R,@#A" TEST WITH [R] = [A]
2951                                  ;;*****************************************************************
2952  006274                          TST64:
2953  006274  012700  000064                  MOV     #64,R0          ;;LOAD R0 WITH TEST NUMBER
2954                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT1
2955  006300  032737  000002  063234          BIT     #BIT1,@#BPTLOC  ;BREAKPOINT HALT SET ??
2956  006306  001401                          BEQ     .+4             ;BR IF NOT
2957  006310  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
2958  006312  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2959  006316  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
2960  006322  010405                  1$:     MOV     R4,R5           ;[R5] = SOURCE OP = 125252
2961  006324  010412                          MOV     R4,(R2)         ;MAKE [DEST] = 125252
2962  006326  000257                          CCC                     ;SCOPE SYNC
2963
2964  006330  020537  063312          2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2965
2966  006334  001402                          BEQ     4$              ;BR IF "Z" WAS SET - IT SHOULD BE
2967
2968  006336  000000                  3$:     HALT                    ;CMP FAILED TO SET "Z"
2969  006340  000770                          BR      1$              ;LOCK ON HARD ERROR
2970
2971  006342  020412                  4$:     CMP     R4,(R2)         ;IS RESULT STILL = 125252 ?
2972  006344  001402                          BEQ     TST65           ;;BR IF YES
2973
2974  006346  000000                  5$:     HALT                    ;CMP ALTERED [DEST]
2975  006350  000764                          BR      1$              ;LOCK ON HARD ERROR
2976
2977                                  ;;*****************************************************************
2978                                  ;*TEST 65        BASIC "CMP  R,@#A" WITH [R] NOT EQUAL TO [A]
2979                                  ;;*****************************************************************
2980  006352                          TST65:
2981  006352  012700  000065                  MOV     #65,R0          ;;LOAD R0 WITH TEST NUMBER
2982  006356  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
2983  006362  012704  125252                  MOV     #125252,R4      ;MAKE RESULT S / B = 125252
2984  006366  005005                  1$:     CLR     R5              ;[R5] = SOURCE OP = 000000
2985  006370  010412                          MOV     R4,(R2)         ;MAKE [DEST] = 125252
2986  006372  000277                          SCC                     ;SCOPE SYNC - MAKE Z-1
2987
2988  006374  020537  063312          2$:     CMP     R5,@#MBUF0      ;TEST THE CMP
2989
2990  006400  001002                          BNE     TST66           ;;BR IF Z 0 - IT SHOULD BE
```

```
2991
2992  006402  000000                    3$:     HALT                    ;CMP FAILED TO CLEAR "Z"
2993  006404  000770                            BR      1$              ;LOCK ON HARD ERROR
2994
2995                                    ;;*******************************************************************
2996                                    ;*TEST 66        BASIC "BIS #N,@#A" TEST - N=177777,[A]=000000
2997                                    ;;*******************************************************************
2998  006406                            TST66:
2999  006406  012700  000066                    MOV     #66,R0          ;;LOAD R0 WITH TEST NUMBER
3000  006412  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3001  006416  012704  177777                    MOV     #-1,R4          ;RESULT S / B = 177777
3002  006422  005012                    1$:     CLR     (R2)            ;[DEST] = 000000
3003  006424  000257                            CCC                     ;SCOPE SYNC
3004
3005  006426  052737  177777  063312    2$:     BIS     #-1,@#MBUF0     ;TEST THE BIS
3006
3007  006434  020412                            CMP     R4,(R2)         ;RESULT OK?
3008  006436  001402                            BEQ     TST67           ;;BR IF YES
3009
3010  006440  000000                    3$:     HALT                    ;BIS FAILED TO SET ALL BITS IN BITFLG
3011  006442  000767                            BR      1$              ;LOCK ON HARD ERROR
3012
3013                                    ;;*******************************************************************
3014                                    ;*TEST 67        BASIC "BIC  #N,@#A" TEST
3015                                    ;;*******************************************************************
3016  006444                            TST67:
3017  006444  012700  000067                    MOV     #67,R0          ;;LOAD R0 WITH TEST NUMBER
3018  006450  012702  063312                    MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3019  006454  012704  000077                    MOV     #77,R4          ;RESULT S / B = 77
3020  006460  012712  177777            1$:     MOV     #-1,(R2)        ;MAKE [DEST] = 177777
3021  006464  000257                            CCC                     ;SCOPE SYNC
3022
3023  006466  042737  177700  063312    2$:     BIC     #177700,@#MBUF0 ;TEST THE BIC
3024
3025  006474  020412                            CMP     R4,(R2)         ;DID RESULT = 77 ?
3026  006476  001402                            BEQ     TST70           ;;BR IF YES
3027
3028  006500  000000                    3$:     HALT                    ;BIC DELIVERED THE WRONG RESULT
3029  006502  000766                            BR      1$              ;LOCK ON HARD ERROR
3030
3031                                    ;;*******************************************************************
3032                                    ;*TEST 70        BASIC "BIC #N,R" TEST
3033                                    ;;*******************************************************************
3034  006504                            TST70:
3035  006504  012700  000070                    MOV     #70,R0          ;;LOAD R0 WITH TEST NUMBER
3036  006510  005003                    1$:     CLR     R3              ;[DEST] = 177777
3037  006512  005103                            COM     R3
3038  006514  000257                            CCC                     ;SCOPE SYNC
3039
3040  006516  042703  177400            2$:     BIC     #177400,R3      ;TEST THE BIC
3041
3042  006522  022703  000377                    CMP     #377,R3         ;RESULT OK?
3043  006526  001402                            BEQ     TST71           ;;BR IF YES
3044
3045  006530  000000                    3$:     HALT                    ;BIC FAILED TO CLEAR HI-BYTE
3046  006532  000766                            BR      1$              ;LOCK ON HARD ERROR
```

```
3047
3048                                    ;;*******************************************************************
3049                                    ;*TEST 71        BASIC "BIC #N,2(SP)" TEST
3050                                    ;;*******************************************************************
3051   006534                           TST71:
3052   006534  012700  000071                  MOV     #71,R0          ;;LOAD R0 WITH TEST NUMBER
3053   006540  012704  000357                  MOV     #357,R4         ;RESULT S / B = 357
3054   006544  010605                          MOV     SP,R5           ;SAVE SP
3055   006546  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3056   006550  012746  000377                  MOV     #377,-(SP)      ;[DEST] = 377 PUT ON STACK
3057   006554  005746                          TST     -(SP)           ;DECREMENT SP
3058   006556  000257                          CCC                     ;SCOPE SYNC
3059
3060   006560  042766  000020  000002  2$:     BIC     #20,2(SP)       ;TEST THE BIC - CLEAR BIT 4
3061
3062   006566  010602                          MOV     SP,R2           ;[R2] = DEST ADDR
3063   006570  005722                          TST     (R2)+
3064   006572  020412                          CMP     R4,(R2)         ;RESULT = 357?
3065   006574  001402                          BEQ     4$              ;BR IF YES
3066
3067   006576  000000                  3$:     HALT                    ;BIC FAILED TO CLR BIT2 OF DEST
3068   006600  000762                          BR      1$              ;LOCK ON HARD ERROR
3069
3070   006602  010506                  4$:     MOV     R5,SP
3071
3072                                    ;;*******************************************************************
3073                                    ;*TEST 72        BASIC "ADD  #N,RN " TEST
3074                                    ;;*******************************************************************
3075   006604                           TST72:
3076   006604  012700  000072                  MOV     #72,R0          ;;LOAD R0 WITH TEST NUMBER
3077   006610  012703  000002          1$:     MOV     #2,R3           ;MAKE [DEST] = 2
3078   006614  000257                          CCC                     ;SCOPE SYNC
3079
3080   006616  062703  000002          2$:     ADD     #2,R3           ;TEST THE ADD
3081
3082   006622  022703  000004                  CMP     #4,R3           ;RESULT = 4 ?
3083   006626  001402                          BEQ     TST73           ;;BR IF YES
3084
3085   006630  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3086   006632  000766                          BR      1$              ;LOCK ON HARD ERROR
3087
3088                                    ;;*******************************************************************
3089                                    ;*TEST 73        BASIC "ADD  #N,(R) " TEST
3090                                    ;;*******************************************************************
3091   006634                           TST73:
3092   006634  012700  000073                  MOV     #73,R0          ;;LOAD R0 WITH TEST NUMBER
3093   006640  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3094   006644  012704  000004                  MOV     #4,R4           ;RESULT S / B = 4
3095   006650  012712  000002          1$:     MOV     #2,(R2)         ;MAKE [DEST] = 2
3096   006654  000257                          CCC                     ;SCOPE SYNC
3097
3098   006656  062712  000002          2$:     ADD     #2,(R2) ;TEST THE ADD
3099
3100   006662  020412                          CMP     R4,(R2)         ;RESULT = 4 ?
3101   006664  001402                          BEQ     TST74           ;;BR IF YES
3102
```

```
3103  006666  000000                  3$:     HALT                    ;ADD DELIVERED THE WRONG RESULT
3104  006670  000767                          BR      1$              ;LOCK ON HARD ERROR
3105
3106                                  ;;*********************************************************************
3107                                  ;*TEST 74        BASIC "ADD  #N,X(R)" TEST
3108                                  ;;*********************************************************************
3109  006672                          TST74:
3110  006672  012700  000074                MOV     #74,R0          ;;LOAD R0 WITH TEST NUMBER
3111  006676  012704  000002                MOV     #2,R4           ;RESULT S / B = 2
3112  006702  012702  063314                MOV     #MBUF0+2,R2     ;DEST ADDR = MBUF0 + 2
3113  006706  012705  063312          1$:   MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
3114  006712  005012                        CLR     (R2)            ;MAKE [DEST] = 000000
3115  006714  000257                        CCC                     ;SCOPE SYNC
3116
3117  006716  062765  000002  000002  2$:   ADD     #2,2(R5)        ;TEST THE ADD
3118
3119  006724  020412                        CMP     R4,(R2)         ;RESULT = 2 ?
3120  006726  001402                        BEQ     TST75           ;;BR IF YES
3121
3122  006730  000000                  3$:   HALT                    ;ADD DELIVERED THE WRONG RESULT
3123  006732  000765                        BR      1$              ;LOOP ON HARD ERROR
3124
3125                                  ;;*********************************************************************
3126                                  ;*TEST 75        BASIC "CMPB #N,(SP)+" TEST
3127                                  ;;*********************************************************************
3128  006734                          TST75:
3129  006734  012700  000075                MOV     #75,R0          ;;LOAD R0 WITH TEST NUMBER
3130  006740  012704  177400                MOV     #177400,R4      ;RESULT S / B = 177400
3131  006744  010605                        MOV     SP,R5           ;SAVE SP
3132  006746  010602                        MOV     SP,R2           ;SET UP DEST ADDR
3133  006750  005742                        TST     -(R2)           ;R2 CONTAINS DEST ADDR
3134  006752  010506                  1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3135  006754  010446                        MOV     R4,-(SP)        ;MAKE [DEST] = 177400
3136  006756  000257                        CCC                     ;SCOPE SYNC - "Z" - 0
3137
3138  006760  122726  000000          2$:   CMPB    #0,(SP)+        ;TEST THE CMPB
3139
3140  006764  001402                        BEQ     4$              ;BR IF "Z" SET - IT SHOULD BE
3141
3142  006766  000000                  3$:   HALT                    ;CMPB FAILED TO SET "Z"
3143  006770  000770                        BR      1$              ;LOCK ON HARD ERROR
3144
3145  006772  020506                  4$:   CMP     R5,SP           ;DID SP GET UPDATED BY 2?
3146  006774  001402                        BEQ     6$              ;BR IF YES
3147
3148  006776  000000                  5$:   HALT                    ;CMPB FAILED TO UPDATE SP PROPERLY
3149  007000  000764                        BR      1$              ;LOCK ON HARD ERROR
3150
3151  007002  020412                  6$:   CMP     R4,(R2)         ;[DEST] ALTERED?
3152  007004  001402                        BEQ     TST76           ;;BR IF NOT
3153
3154  007006  000000                  7$:   HALT                    ;CMPB MODIFIED [DEST]
3155  007010  000760                        BR      1$              ;LOCK ON HARD ERROR.
3156
3157                                  ;;*********************************************************************
3158                                  ;*TEST 76        BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST EVEN
```

```
3159
3160  007012                          ;;*****************************************************************
3161  007012                          TST76:
3162  007012  012700  000076                 MOV     #76,R0          ;;LOAD R0 WITH TEST NUMBER
3163  007016  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3164  007022  012702  063324                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3165  007026  012705  063330         1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3166  007032  010203                         MOV     R2,R3           ;R3 GETS DEST ADDR
3167  007034  000257                         CCC                     ;SCOPE SYNC
3168
3169  007036  122523                 2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3170
3171  007040  001402                         BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3172
3173  007042  000000                 3$:     HALT                    ;CMPB FAILED TO SET "Z"
3174  007044  000770                         BR      1$              ;LOCK ON HARD ERROR
3175
3176  007046  022703  063325         4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
3177  007052  001402                         BEQ     6$              ;BR IF YES
3178
3179  007054  000000                 5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3180  007056  000763                         BR      1$              ;LOCK ON HARD ERROR
3181
3182  007060  022705  063331         6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3183  007064  001402                         BEQ     8$              ;BR IF YES
3184
3185  007066  000000                 7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3186  007070  000756                         BR      1$              ;LOCK ON HARD ERROR
3187
3188  007072  020412                 8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3189  007074  001403                         BEQ     TST77           ;;BR IF NOT
3190
3191  007076  000000                 9$:     HALT                    ;CMPB DELIVERED A RESULT
3192  007100  010412                         MOV     R4,(R2)         ;RESTORE [DEST]
3193  007102  000751                         BR      1$              ;LOCK ON HARD ERROR
3194
3195                                  ;;*****************************************************************
3196                                  ;*TEST 77       BASIC "CMPB (RA)+,(RB)+" - SRC AND DEST ODD
3197                                  ;;*****************************************************************
3198  007104                          TST77:
3199  007104  012700  000077                 MOV     #77,R0          ;;LOAD R0 WITH TEST NUMBER
3200  007110  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
3201  007114  012702  063324                 MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3202  007120  012705  063327         1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
3203  007124  012703  063325                 MOV     #DWTA+3,R3      ;R3 GETS DEST ADDR+1
3204  007130  000257                         CCC                     ;SCOPE SYNC
3205
3206  007132  122523                 2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3207
3208  007134  001402                         BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3209
3210  007136  000000                 3$:     HALT                    ;CMPB FAILED TO SET "Z"
3211  007140  000767                         BR      1$              ;LOCK ON HARD ERROR
3212
3213  007142  022703  063326         4$:     CMP     #DWTA+4,R3      ;DID DEST REG GET UPDATED?
3214  007146  001402                         BEQ     6$              ;BR IF YES
```

```
                                                                L 5
3215  007150  000000              5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3216  007152  000762                      BR      1$              ;LOCK ON HARD ERROR
3217
3218  007154  022705  063330      6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3219  007160  001402                      BEQ     8$              ;BR IF YES
3220
3221  007162  000000              7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3222  007164  000755                      BR      1$              ;LOCK ON HARD ERROR
3223
3224  007166  020412              8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3225  007170  001403                      BEQ     TST100          ;;BR IF NOT
3226
3227  007172  000000              9$:     HALT                    ;CMPB DELIVERED A RESULT
3228  007174  010412                      MOV     R4,(R2)         ;RESTORE [DEST]
3229  007176  000750                      BR      1$              ;LOCK ON HARD ERROR
3230
3231                              ;;*********************************************************************
3232                              ;*TEST 100      BASIC "CMPB (RA)+,(RB)+" - SRC / EVEN,DEST / ODD
3233                              ;;*********************************************************************
3234  007200              TST100:
3235  007200  012700  000100              MOV     #100,R0         ;;LOAD R0 WITH TEST NUMBER
3236  007204  012704  177400              MOV     #177400,R4      ;RESULT S / B = 177400
3237  007210  012702  063326              MOV     #DWTA+4,R2      ;DEST ADDR = DWTA+4
3238  007214  012705  063330      1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3239  007220  012703  063327              MOV     #DWTA+5,R3      ;R3 GETS DEST ADDR
3240  007224  000257                      CCC                     ;SCOPE SYNC
3241
3242  007226  122523              2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3243
3244  007230  001402                      BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3245
3246  007232  000000              3$:     HALT                    ;CMPB FAILED TO SET "Z"
3247  007234  000767                      BR      1$              ;LOCK ON HARD ERROR
3248
3249  007236  022703  063330      4$:     CMP     #DWTA+6,R3      ;DID DEST REG GET UPDATED?
3250  007242  001402                      BEQ     6$              ;BR IF YES
3251
3252  007244  000000              5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3253  007246  000762                      BR      1$              ;LOCK ON HARD ERROR
3254
3255  007250  022705  063331      6$:     CMP     #DWTA+7,R5      ;DID SRC REG GET UPDATED?
3256  007254  001402                      BEQ     8$              ;BR IF YES
3257
3258  007256  000000              7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3259  007260  000755                      BR      1$              ;LOCK ON HARD ERROR
3260
3261  007262  020412              8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3262  007264  001403                      BEQ     TST101          ;;BR IF NOT
3263
3264  007266  000000              9$:     HALT                    ;CMPB DELIVERED A RESULT
3265  007270  010412                      MOV     R4,(R2)         ;RESTORE [DEST]
3266  007272  000750                      BR      1$              ;LOCK ON HARD ERROR
3267
3268                              ;;*********************************************************************
3269                              ;*TEST 101      BASIC "CMPB (RA)+,(RB)+" - SRC / ODD,DEST / EVEN
3270                              ;;*********************************************************************
```

```
3271  007274                          TST101:
3272  007274  012700  000101                  MOV     #101,R0         ;;LOAD R0 WITH TEST NUMBER
3273  007300  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
3274  007304  012702  063324                  MOV     #DWTA+2,R2      ;DEST ADDR = DWTA+2
3275  007310  012705  063327          1$:     MOV     #DWTA+5,R5      ;SRC ADDR = DWTA+5
3276  007314  010203                          MOV     R2,R3           ;R3 GETS DEST ADDR
3277  007316  000257                          CCC                     ;SCOPE SYNC
3278
3279  007320  122523                  2$:     CMPB    (R5)+,(R3)+     ;TEST THE CMPB
3280
3281  007322  001402                          BEQ     4$              ;BR IF "Z" = 1 - IT SHOULD BE
3282
3283  007324  000000                  3$:     HALT                    ;CMPB FAILED TO SET "Z"
3284  007326  000770                          BR      1$              ;LOCK ON HARD ERROR
3285
3286  007330  022703  063325          4$:     CMP     #DWTA+3,R3      ;DID DEST REG GET UPDATED?
3287  007334  001402                          BEQ     6$              ;BR IF YES
3288
3289  007336  000000                  5$:     HALT                    ;CMPB FAILED TO UPDATE DEST REG
3290  007340  000763                          BR      1$              ;LOCK ON HARD ERROR
3291
3292  007342  022705  063330          6$:     CMP     #DWTA+6,R5      ;DID SRC REG GET UPDATED?
3293  007346  001402                          BEQ     8$              ;BR IF YES
3294
3295  007350  000000                  7$:     HALT                    ;CMPB FAILED TO UPDATE SRC REG
3296  007352  000756                          BR      1$              ;LOCK ON HARD ERROR
3297
3298  007354  020412                  8$:     CMP     R4,(R2)         ;DID [DEST] GET ALTERED?
3299  007356  001403                          BEQ     TST102          ;;BR IF NOT
3300
3301  007360  000000                  9$:     HALT                    ;CMPB DELIVERED A RESULT
3302  007362  010412                          MOV     R4,(R2)         ;RESTORE [DEST]
3303  007364  000751                          BR      1$              ;LOCK ON HARD ERROR
3304
3305                                  ;;*******************************************************************
3306                                  ;*TEST 102       BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST EVEN
3307                                  ;;*******************************************************************
3308  007366                          TST102:
3309  007366  012700  000102                  MOV     #102,R0         ;;LOAD R0 WITH TEST NUMBER
3310  007372  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3311  007376  012703  063312                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3312  007402  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
3313  007406  012705  064630          1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
3314  007412  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
3315  007416  000257                          CCC                     ;SCOPE SYNC
3316
3317  007420  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3318
3319  007424  020412                          CMP     R4,(R2)         ;RESULT OK?
3320  007426  001402                          BEQ     4$              ;BR IF YES
3321
3322  007430  000000                  3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3323  007432  000765                          BR      1$              ;LOCK ON HARD ERROR
3324
3325  007434  022705  064631          4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
3326  007440  001402                          BEQ     TST103          ;;BR IF YES
```

```
3327
3328  007442  000000            5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3329  007444  000760                    BR      1$              ;LOCK ON HARD ERROR
3330                            ;;*********************************************************************
3331                            ;*TEST 103      BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST ODD
3332                            ;;*********************************************************************
3333  007446                    TST103:
3334  007446  012700  000103            MOV     #103,R0         ;;LOAD R0 WITH TEST NUMBER
3335  007452  012702  063316            MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3336  007456  012703  063312            MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3337  007462  012704  000777            MOV     #777,R4         ;RESULT S / B = 777
3338  007466  012705  064635    1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3339  007472  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
3340  007476  000257                    CCC                     ;SCOPE SYNC
3341
3342  007500  112563  000005    2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3343
3344  007504  020412                    CMP     R4,(R2)         ;RESULT OK?
3345  007506  001402                    BEQ     4$              ;BR IF YES
3346
3347  007510  000000            3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3348  007512  000765                    BR      1$              ;LOCK ON HARD ERROR
3349
3350  007514  022705  064636    4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3351  007520  001402                    BEQ     TST104          ;;BR IF YES
3352
3353  007522  000000            5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3354  007524  000760                    BR      1$              ;LOCK ON HARD ERROR
3355                            ;;*********************************************************************
3356                            ;*TEST 104      BASIC "MOVB (RA)+,X(RB) - SRC EVEN / DEST ODD
3357                            ;;*********************************************************************
3358  007526                    TST104:
3359  007526  012700  000104            MOV     #104,R0         ;;LOAD R0 WITH TEST NUMBER
3360  007532  012702  063316            MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3361  007536  012703  063312            MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3362  007542  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
3363  007546  012705  064630    1$:     MOV     #DBTA,R5        ;SRC ADDR = DBTA
3364  007552  012712  177777            MOV     #-1,(R2)        ;[DEST] = 177777
3365  007556  000257                    CCC                     ;SCOPE SYNC
3366
3367  007560  112563  000005    2$:     MOVB    (R5)+,5(R3)     ;TEST THE MOVB
3368
3369  007564  020412                    CMP     R4,(R2)         ;RESULT OK?
3370  007566  001402                    BEQ     4$              ;BR IF YES
3371
3372  007570  000000            3$:     HALT                    ;MOV DELIVERED WRONG RESULT
3373  007572  000765                    BR      1$              ;LOCK ON HARD ERROR
3374
3375  007574  022705  064631    4$:     CMP     #DBTA+1,R5      ;DID SRC REG GET INCREMENTED BY +1
3376  007600  001402                    BEQ     TST105          ;;BR IF YES
3377
3378  007602  000000            5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3379  007604  000760                    BR      1$              ;LOCK ON HARD ERROR
3380                            ;;*********************************************************************
3381                            ;*TEST 105      BASIC "MOVB (RA)+,X(RB) - SRC ODD / DEST EVEN
3382                            ;;*********************************************************************
```

```
3383  007606                          TST105:
3384  007606  012700  000105                  MOV     #105,R0         ;;LOAD R0 WITH TEST NUMBER
3385  007612  012702  063316                  MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
3386  007616  012703  063312                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
3387  007622  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3388  007626  012705  064635          1$:     MOV     #DBTB+1,R5      ;SRC ADDR = DBTB+1
3389  007632  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
3390  007636  000257                          CCC                     ;SCOPE SYNC
3391
3392  007640  112563  000004          2$:     MOVB    (R5)+,4(R3)     ;TEST THE MOVB
3393
3394  007644  020412                          CMP     R4,(R2)         ;RESULT OK?
3395  007646  001402                          BEQ     4$              ;BR IF YES
3396
3397  007650  000000          3$:     HALT                            ;MOV DELIVERED WRONG RESULT
3398  007652  000765                          BR      1$              ;LOCK ON HARD ERROR
3399
3400  007654  022705  064636          4$:     CMP     #DBTB+2,R5      ;DID SRC REG GET INCREMENTED BY +1
3401  007660  001402                          BEQ     TST106          ;;BR IF YES
3402
3403  007662  000000          5$:     HALT                            ;MOVB FAILED TO UPDATE SRC REG
3404  007664  000760                          BR      1$              ;LOCK ON HARD ERROR
3405
3406                                  ;;*******************************************************************
3407                                  ;*TEST 106       BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST EVEN
3408                                  ;;*******************************************************************
3409  007666                          TST106:
3410  007666  012700  000106                  MOV     #106,R0         ;;LOAD R0 WITH TEST NUMBER
3411  007672  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3412  007676  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
3413  007702  012705  064032                  MOV     #DWTB,R5        ;SRC ADDR = DWTB
3414  007706  010203          1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3415  007710  012713  177777                  MOV     #-1,(R3)        ;[DEST] = 177400
3416  007714  000257                          CCC                     ;SCOPE SYNC
3417
3418  007716  116523  000002          2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3419
3420  007722  020412                          CMP     R4,(R2)         ;RESULT OK?
3421  007724  001402                          BEQ     4$              ;BR IF YES
3422
3423  007726  000000          3$:     HALT                            ;MOVB DELIVERED WRONG RESULT
3424  007730  000766                          BR      1$              ;LOCK ON HARD ERROR
3425
3426  007732  022703  063313          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3427  007736  001402                          BEQ     TST107          ;;BR IF YES
3428
3429  007740  000000          5$:     HALT                            ;MOVB FAILED TO AUTO INCREMENT DEST REG
3430  007742  000761                          BR      1$              ;LOCK ON HARD ERROR
3431
3432                                  ;;*******************************************************************
3433                                  ;*TEST 107       BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST EVEN
3434                                  ;;*******************************************************************
3435  007744                          TST107:
3436  007744  012700  000107                  MOV     #107,R0         ;;LOAD R0 WITH TEST NUMBER
3437  007750  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3438  007754  012704  177401                  MOV     #177401,R4      ;RESULT S / B = 177401
```

```
3439  007760  012705  064634          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3440  007764  010203          1$:     MOV     R2,R3           ;R3 GETS DEST ADDR
3441  007766  012713  177777          MOV     #-1,(R3)        ;[DEST] = 177777
3442  007772  000257                  CCC                     ;SCOPE SYNC
3443
3444  007774  116523  000001  2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3445
3446  010000  020412                  CMP     R4,(R2)         ;RESULT OK?
3447  010002  001402                  BEQ     4$              ;BR IF YES
3448
3449  010004  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3450  010006  000766                  BR      1$              ;LOCK ON HARD ERROR
3451
3452  010010  022703  063313  4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED?
3453  010014  001402                  BEQ     TST110          ;;BR IF YES
3454
3455  010016  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3456  010020  000761                  BR      1$              ;LOCK ON HARD ERROR
3457
3458                          ;;**********************************************************************
3459                          ;*TEST 110       BASIC "MOVB 2(RA),(RB)+" TEST - SRC EVEN / DEST ODD
3460                          ;;**********************************************************************
3461  010022          TST110:
3462  010022  012700  000110          MOV     #110,R0         ;;LOAD R0 WITH TEST NUMBER
3463  010026  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3464  010032  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3465  010036  012705  064032          MOV     #DWTB,R5        ;SRC ADDR = DWTB
3466  010042  012703  063313  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR
3467  010046  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3468  010052  000257                  CCC                     ;SCOPE SYNC
3469
3470  010054  116523  000002  2$:     MOVB    2(R5),(R3)+     ;TEST THE MOVB
3471
3472  010060  020412                  CMP     R4,(R2)         ;RESULT OK?
3473  010062  001402                  BEQ     4$              ;BR IF YES
3474
3475  010064  000000          3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3476  010066  000765                  BR      1$              ;LOCK ON HARD ERROR
3477
3478  010070  022703  063314  4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3479  010074  001402                  BEQ     TST111          ;;BR IF YES
3480
3481  010076  000000          5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3482  010100  000760                  BR      1$              ;LOCK ON HARD ERROR
3483
3484                          ;;**********************************************************************
3485                          ;*TEST 111       BASIC "MOVB 2(RA),(RB)+" TEST - SRC ODD / DEST ODD
3486                          ;;**********************************************************************
3487  010102          TST111:
3488  010102  012700  000111          MOV     #111,R0         ;;LOAD R0 WITH TEST NUMBER
3489  010106  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3490  010112  012704  000777          MOV     #777,R4         ;RESULT S / B = 777
3491  010116  012705  064634          MOV     #DBTB,R5        ;SRC ADDR = DBTB
3492  010122  012703  063313  1$:     MOV     #MBUF0+1,R3     ;R3 GETS DEST ADDR - MBUF0+1
3493  010126  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
3494  010132  000257                  CCC                     ;SCOPE SYNC
```

```
3495
3496  010134  116523  000001        2$:     MOVB    1(R5),(R3)+     ;TEST THE MOVB
3497
3498  010140  020412                        CMP     R4,(R2)         ;RESULT OK?
3499  010142  001402                        BEQ     4$              ;BR IF YES
3500
3501  010144  000000                3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3502  010146  000765                        BR      1$              ;LOCK ON HARD ERROR
3503
3504  010150  022703  063314        4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED?
3505  010154  001402                        BEQ     TST112          ;;BR IF YES
3506
3507  010156  000000                5$:     HALT                    ;MOVB FAILED TO AUTO INCREMENT DEST REG
3508  010160  000760                        BR      1$              ;LOCK ON HARD ERROR
3509
3510                                ;;**************************************************************
3511                                ;*TEST 112       BASIC "MOVB -(RA),RB" TEST - SRC EVEN ADDR
3512                                ;;**************************************************************
3513  010162                        TST112:
3514  010162  012700  000112                MOV     #112,R0         ;;LOAD R0 WITH TEST NUMBER
3515  010166  012705  063331        1$:     MOV     #DWTA+7,R5      ;SRC ADDR = DWTA+7
3516  010172  005003                        CLR     R3              ;[DEST] = 000000
3517  010174  000257                        CCC                     ;SCOPE SYNC
3518
3519  010176  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3520
3521  010200  022703  177777                CMP     #-1,R3          ;RESULT OK?
3522  010204  001402                        BEQ     4$              ;BR IF YES
3523
3524  010206  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3525  010210  000766                        BR      1$              ;LOCK ON HARD ERROR
3526
3527  010212  022705  063330        4$:     CMP     #DWTA+6,R5      ;SRC REG GET DECREMENTED?
3528  010216  001402                        BEQ     TST113          ;;BR IF YES
3529
3530  010220  000000                5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3531  010222  000761                        BR      1$              ;LOCK ON HARD ERROR
3532                                ;;**************************************************************
3533                                ;*TEST 113       BASIC "MOVB -(RA),RB" TEST - SRC ODD ADDR
3534                                ;;**************************************************************
3535  010224                        TST113:
3536  010224  012700  000113                MOV     #113,R0         ;;LOAD R0 WITH TEST NUMBER
3537  010230  012705  063330        1$:     MOV     #DWTA+6,R5      ;SRC ADDR = DWTA+6
3538  010234  005003                        CLR     R3              ;[DEST] = 000000
3539  010236  000257                        CCC                     ;SCOPE SYNC
3540
3541  010240  114503                2$:     MOVB    -(R5),R3        ;TEST THE MOVB
3542
3543  010242  022703  177777                CMP     #-1,R3          ;RESULT OK?
3544  010246  001402                        BEQ     4$              ;BR IF YES
3545
3546  010250  000000                3$:     HALT                    ;MOVB FAILED - WRONG RESULT
3547  010252  000766                        BR      1$              ;LOCK ON HARD ERROR
3548
3549  010254  022705  063327        4$:     CMP     #DWTA+5,R5      ;SRC REG GET DECREMENTED?
3550  010260  001402                        BEQ     TST114          ;;BR IF YES
```

```
3551
3552  010262  000000                  5$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3553  010264  000761                          BR      1$              ;LOCK ON HARD ERROR
3554
3555                                  ;;*********************************************************************
3556                                  ;*TEST 114      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR EVEN
3557                                  ;;*********************************************************************
3558  010266                          TST114:
3559  010266  012700  000114                 MOV     #114,R0         ;;LOAD R0 WITH TEST NUMBER
3560  010272  010605                          MOV     SP,R5           ;SAVE SP
3561  010274  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
3562  010300  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3563  010302  012703  064630                  MOV     #DBTA,R3        ;SRC ADDR = DBTA
3564  010306  012746  177777                  MOV     #-1,-(SP)       ;[DEST] = 177777
3565  010312  010602                          MOV     SP,R2           ;R2 GETS DEST ADDR
3566  010314  005726                          TST     (SP)+           ;RESET SP
3567  010316  000257                          CCC                     ;SCOPE SYNC
3568
3569  010320  112346                  2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3570
3571  010322  022703  064631                  CMP     #DBTA+1,R3      ;DID MOVB INCREMENT SRC REG?
3572  010326  001402                          BEQ     4$              ;BR IF YES
3573
3574  010330  000000                  3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3575  010332  000762                          BR      1$              ;LOCK ON HARD ERROR
3576
3577  010334  020412                  4$:     CMP     R4,(R2)         ;RESULT OK?
3578  010336  001402                          BEQ     6$              ;BR IF YES
3579
3580  010340  000000                          HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3581  010342  000756                          BR      1$              ;LOCK ON HARD ERROR
3582
3583  010344  020206                  6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2 ?
3584  010346  001402                          BEQ     8$              ;BR IF YES
3585
3586  010350  000000                          HALT                    ;MOVB FAILED TO PUSH SP PROPERLY
3587  010352  000752                          BR      1$              ;LOCK ON HARD ERROR
3588
3589  010354  010506                  8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3590
3591                                  ;;*********************************************************************
3592                                  ;*TEST 115      BASIC "MOVB (RA)+,-(SP)" TEST - SRC ADDR ODD
3593                                  ;;*********************************************************************
3594  010356                          TST115:
3595  010356  012700  000115                 MOV     #115,R0         ;;LOAD R0 WITH TEST NUMBER
3596  010362  010605                          MOV     SP,R5           ;SAVE SP
3597  010364  012704  177400                  MOV     #177400,R4      ;RESULT S / B = 177400
3598  010370  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
3599  010372  012703  064035                  MOV     #DWTB+3,R3      ;SRC ADDR = DWTB+3
3600  010376  012746  177777                  MOV     #-1,-(SP)       ;[DEST] = 177777
3601  010402  010602                          MOV     SP,R2           ;R2 GETS DEST ADDR
3602  010404  005726                          TST     (SP)+           ;RESET SP
3603  010406  000257                          CCC                     ;SCOPE SYNC
3604
3605  010410  112346                  2$:     MOVB    (R3)+,-(SP)     ;TEST THE MOVB
3606
```

```
3607  010412  022703  064036          CMP     #DWTB+4,R3      ;DID MOVB INCREMENT SRC REG?
3608  010416  001402                  BEQ     4$              ;BR IF YES
3609
3610  010420  000000          3$:     HALT                    ;MOVB FAILED TO UPDATE SRC REG
3611  010422  000762                  BR      1$              ;LOCK ON HARD ERROR
3612
3613  010424  020412          4$:     CMP     R4,(R2)         ;RESULT OK?
3614  010426  001402                  BEQ     6$              ;BR IF YES
3615
3616  010430  000000                  HALT                    ;MOVB FAILED TO DELIVER CORRECT RESULT
3617  010432  000756                  BR      1$              ;LOCK ON HARD ERROR
3618
3619  010434  020206          6$:     CMP     R2,SP           ;DID SP GET PUSHED BY 2
3620  010436  001402                  BEQ     8$              ;BR IF YES
3621
3622  010440  000000                  HALT                    ;MOVB FAILED TO PUSH SP
3623  010442  000752                  BR      1$              ;LOCK ON HARD ERROR
3624
3625  010444  010506          8$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
3626
3627                          ;;*******************************************************************
3628                          ;*TEST 116       BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST EVEN
3629                          ;;*******************************************************************
3630  010446                  TST116:
3631  010446  012700  000116          MOV     #116,R0         ;,LOAD R0 WITH TEST NUMBER
3632  010452  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3633  010456  012704  000001          MOV     #1,R4           ;RESULT S / B - 1
3634  010462  012705  064032          MOV     #DWTB,R5        ;BASE SRC ADDR   DWTB
3635  010466  005012          1$:     CLR     (R2)            ;[DEST] = 000000
3636  010470  000257                  CCC                     ;SCOPE SYNC
3637
```

```
3638  010472  116537  000006  063312  2$:     MOVB    6(R5),@#MBUF0   ;TEST THE MOVB
3639
3640  010500  020412                          CMP     R4,(R2)         ;RESULT OK?
3641  010502  001402                          BEQ     TST117          ;;BR IF YES
3642
3643  010504  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3644  010506  000767                          BR      1$              ;LOCK ON HARD ERRROR
3645                                  ;;*******************************************************
3646                                  ;*TEST 117      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST EVEN
3647                                  ;;*******************************************************
3648  010510                          TST117:
3649  010510  012700  000117                  MOV     #117,R0         ;;LOAD R0 WITH TEST NUMBER
3650  010514  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3651  010520  012704  000001                  MOV     #1,R4           ;RESULT S / B = 1
3652  010524  012705  064634                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3653  010530  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3654  010532  000257                          CCC                     ;SCOPE SYNC
3655
3656  010534  116537  000001  063312  2$:     MOVB    1(R5),@#MBUF0   ;TEST THE MOVB
3657
3658  010542  020412                          CMP     R4,(R2)         ;RESULT OK?
3659  010544  001402                          BEQ     TST120          ;;BR IF YES
3660
3661  010546  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3662  010550  000767                          BR      1$              ;LOCK ON HARD ERRROR
3663                                  ;;*******************************************************
3664                                  ;*TEST 120      BASIC "MOVB X(R),@#A" TEST - SRC EVEN / DEST ODD
3665                                  ;;*******************************************************
3666  010552                          TST120:
3667  010552  012700  000120                  MOV     #120,R0         ;;LOAD R0 WITH TEST NUMBER
3668  010556  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
3669  010562  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3670  010566  012705  064032                  MOV     #DWTB,R5        ;BASE SRC ADDR - DWTB
3671  010572  005012                  1$:     CLR     (R2)            ;[DEST] = 000000
3672  010574  000257                          CCC                     ;SCOPE SYNC
3673
3674  010576  116537  000006  063313  2$:     MOVB    6(R5),@#MBUF0+1 ;TEST THE MOVB
3675
3676  010604  020412                          CMP     R4,(R2)         ;RESULT OK?
3677  010606  001402                          BEQ     TST121          ;;BR IF YES
3678
3679  010610  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3680  010612  000767                          BR      1$              ;LOCK ON HARD ERRROR
3681                                  ;;*******************************************************
3682                                  ;*TEST 121      BASIC "MOVB X(R),@#A" TEST - SRC ODD / DEST ODD
3683                                  ;;*******************************************************
3684  010614                          TST121:
3685  010614  012700  000121                  MOV     #121,R0         ;;LOAD R0 WITH TEST NUMBER
3686                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT2
3687  010620  032737  000004  063234          BIT     #BIT2,@#BPTLOC  ;BREAKPOINT HALT SET ??
3688  010626  001401                          BEQ     .+4             ;BR IF NOT
3689  010630  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
3690  010632  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR - MBUF0
3691  010636  012704  000400                  MOV     #400,R4         ;RESULT S / B = 400
3692  010642  012705  064634                  MOV     #DBTB,R5        ;BASE SRC ADDR = DBTB
3693  010646  005012                  1$:     CLR     (R2)            ;[DEST]   000000
```

```
3694  010650  000257                          CCC                     ;SCOPE SYNC
3695
3696  010652  116537  000001  063313  2$:     MOVB    1(R5),@#MBUF0+1 ;TEST THE MOVB
3697
3698  010660  020412                          CMP     R4,(R2)         ;RESULT OK?
3699  010662  001402                          BEQ     TST122          ;;BR IF YES
3700
3701  010664  000000                  3$:     HALT                    ;MOVB DELIVERED WRONG RESULT
3702  010666  000767                          BR      1$              ;LOCK ON HARD ERRROR
3703
3704                                  ;;**********************************************************
3705                                  ;*TEST 122      BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLG 0
3706                                  ;;**********************************************************
3707  010670                          TST122:
3708  010670  012700  000122                  MOV     #122,R0         ;;LOAD R0 WITH TEST NUMBER
3709  010674  000257                  1$:     CCC                     ;CLEAR ALL FLAGS
3710
3711  010676  001404                  2$:     BEQ     3$              ;NO BR SHOULD OCCUR-FLAG-0
3712  010700  100403                          BMI     3$              ;NO BR SHOULD OCCUR-FLAG-0
3713  010702  102402                          BVS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3714  010704  103401                          BCS     3$              ;NO BR SHOULD OCCUR-FLAG=0
3715  010706  000402                          BR      TST123          ;;GO TO NEXT TEST
3716
3717  010710  000000                  3$:     HALT                    ;ONE OF ABOVE BR'S FAILED
3718  010712  000770                          BR      1$              ;ERROR LOOP RETURN
3719
3720                                  ;;**********************************************************
3721                                  ;*TEST 123      BASIC QUICK VERIFY TEST FOR BMI,BEQ,BVS,BCS-FLAG 1
3722                                  ;;**********************************************************
3723  010714                          TST123:
3724  010714  012700  000123                  MOV     #123,R0         ;;LOAD R0 WITH TEST NUMBER
3725  010720  000277                  1$:     SCC                     ;MAKE N:C = 1111
3726
3727  010722  001402                  21$:    BEQ     22$             ;TEST THE BEQ-IT SHOULD BR
3728
3729  010724  000000                  3$:     HALT                    ;BEQ FAILED
3730  010726  000774                          BR      1$              ;ERROR LOOP RETURN
3731
3732  010730  100402                  22$:    BMI     23$             ;TEST THE BMI-IT SHOULD BR
3733
3734  010732  000000                  5$:     HALT                    ;BMI FAILED
3735  010734  000771                          BR      1$              ;ERROR LOOP RETURN
3736
3737  010736  102402                  23$:    BVS     24$             ;TEST THE BVS-IT SHOULD BR
3738
3739  010740  000000                  7$:     HALT                    ;BVS FAILED
3740  010742  000766                          BR      1$              ;ERROR LOOP RETURN
3741
3742  010744                          24$:
3743  010744  103402                          BCS     TST124          ;;TEST THE BCS-IT SHOULD BR
3744
3745  010746  000000                  9$:     HALT                    ;BCS FAILED
3746  010750  000763                          BR      1$              ;ERROR LOOP RETURN
3747
3748                                  ;;**********************************************************
3749                                  ;*TEST 124      BASIC BVC TEST WITH V=1
```

```
3750                                    ;;*****************************************************************
3751  010752                            TST124:
3752  010752  012700  000124                    MOV     #124,R0         ;;LOAD R0 WITH TEST NUMBER
3753
3754  010756  000262                    1$:     SEV                     ;MAKE V=1
3755
3756  010760  102001                    2$:     BVC     3$              ;TEST THE BVC-IT SHOULDN'T BR
3757  010762  000402                            BR      TST125          ;;GO TO NEXT TEST
3758
3759  010764  000000                    3$:     HALT                    ;BVC FAILED
3760  010766  000773                            BR      1$              ;ERROR LOOP RETURN
3761
3762                                    ;;*****************************************************************
3763                                    ;*TEST 125      BASIC BVC TEST WITH V=0
3764                                    ;;*****************************************************************
3765  010770                            TST125:
3766  010770  012700  000125                    MOV     #125,R0         ;;LOAD R0 WITH TEST NUMBER
3767
3768  010774  000242                    1$:     CLV                     ;MAKE V=0
3769
3770  010776                            2$:
3771  010776  102002                            BVC     TST126          ;;TEST THE BVC-IT SHOULD BR
3772
3773  011000  000000                    3$:     HALT                    ;BVC FAILED
3774  011002  000774                            BR      1$              ;ERROR LOOP RETURN
3775
3776                                    ;;*****************************************************************
3777                                    ;*TEST 126      BASIC BGE TEST WITH N,V = 00
3778                                    ;;*****************************************************************
3779  011004                            TST126:
3780  011004  012700  000126                    MOV     #126,R0         ;;LOAD R0 WITH TEST NUMBER
3781
3782  011010  000257                    1$:     CCC                     ;MAKE N:C = 0000
3783
3784  011012                            2$:
3785  011012  002002                            BGE     TST127          ;;TEST THE BGE-IT SHOULD BR
3786
3787  011014  000000                    3$:     HALT                    ;BGE FAILED
3788  011016  000774                            BR      1$              ;ERROR LOOP RETURN
3789
3790                                    ;;*****************************************************************
3791                                    ;*TEST 127      BASIC BGE TEST WITH N,V = 01
3792                                    ;;*****************************************************************
3793  011020                            TST127:
3794  011020  012700  000127                    MOV     #127,R0         ;;LOAD R0 WITH TEST NUMBER
3795
3796  011024  000257                    1$:     CCC                     ;CLEAR FLAGS
3797  011026  000262                            SEV                     ;MAKE N,V = 01
3798
3799  011030  002001                    2$:     BGE     3$              ;TEST THE BGE-IT SHOULDN'T BR
3800  011032  000402                            BR      TST130          ;;GO TO NEXT TEST
3801
3802  011034  000000                    3$:     HALT                    ;BGE FAILED
3803  011036  000772                            BR      1$              ;ERROR LOOP RETURN
3804
3805                                    ;;*****************************************************************
```

```
                                ;*TEST 130      BASIC BGE TEST WITH N,V = 10
                                ;;*****************************************************************
                                TST130:
011040                          MOV     #130,R0         ;;LOAD R0 WITH TEST NUMBER
011040  012700  000130
                                1$:     CCC             ;CLEAR FLAGS
011044  000257                  SEN                     ;MAKE N,V = 10
011046  000270
                                2$:     BGE     3$      ;TEST THE BGE-IT SHOULDN'T BR
011050  002001                  BR      TST131          ;;GO TO NEXT TEST
011052  000402
                                3$:     HALT            ;BGE FAILED
011054  000000                  BR      1$              ;ERROR LOOP RETURN
011056  000772
                                ;;*****************************************************************
                                ;*TEST 131      BASIC BGE TEST WITH N,V = 11
                                ;;*****************************************************************
                                TST131:
011060                          MOV     #131,R0         ;;LOAD R0 WITH TEST NUMBER
011060  012700  000131
                                1$:     CCC             ;CLEAR FLAGS
011064  000257                  272                     ;MAKE N,V = 11
011066  000272
                                2$:
011070                          BGE     TST132          ;;TEST THE BGE-IT SHOULD BR
011070  002002
                                3$:     HALT            ;BGE FAILED
011072  000000                  BR      1$              ;ERROR LOOP RETURN
011074  000773
                                ;;*****************************************************************
                                ;*TEST 132      BASIC BLT TEST WITH N,V = 00
                                ;;*****************************************************************
                                TST132:
011076                          MOV     #132,R0         ;;LOAD R0 WITH TEST NUMBER
011076  012700  000132
                                1$:     CCC             ;CLEAR FLAGS
011102  000257
                                2$:     BLT     3$      ;TEST THE BLT-IT SHOULDN'T BR
011104  002401                  BR      TST133          ;;GO TO NEXT TEST
011106  000402
                                3$:     HALT            ;BLT FAILED
011110  000000                  BR      1$              ;ERROR LOOP RETURN
011112  000773
                                ;;*****************************************************************
                                ;*TEST 133      BASIC BLT TEST WITH N,V = 01
                                ;;*****************************************************************
                                TST133:
011114                          MOV     #133,R0         ;;LOAD R0 WITH TEST NUMBER
011114  012700  000133
                                1$:     CCC             ;CLEAR FLAGS
011120  000257                  SEV                     ;MAKE N,V = 01
011122  000262
                                2$:
011124                          BLT     TST134          ;;TEST THE BLT-IT SHOULD BR
011124  002402
                                3$:     HALT            ;BLT FAILED
011126  000000
```

```
3862  011130  000773                    BR      1$              ;ERROR LOOP RETURN
3863
3864                                    ;;*****************************************************************
3865                                    ;*TEST 134      BASIC BLT TEST WITH N,V = 10
3866                                    ;;*****************************************************************
3867  011 32                            TST134:
3868  011132  012700  000134            MOV     #134,R0         ;;LOAD R0 WITH TEST NUMBER
3869
3870  011136  000257                    1$:     CCC             ;CLEAR FLAGS
3871  011140  000270                    SEN                     ;SET N - N,V = 10
3872
3873  011142                            2$:
3874  011142  002402                    BLT     TST135          ;;TEST THE BLT-IT SHOULD BR
3875
3876  011144  000000                    3$:     HALT            ;BLT FAILED
3877  011146  000773                    BR      1$              ;ERROR LOOP RETURN
3878
3879                                    ;;*****************************************************************
3880                                    ;*TEST 135      BASIC BLT TEST WITH N,V = 11
3881                                    ;;*****************************************************************
3882  011150                            TST135:
3883  011150  012700  000135            MOV     #135,R0         ;;LOAD R0 WITH TEST NUMBER
3884
3885  011154  000257                    1$:     CCC             ;CLEAR FLAGS
3886  011156  000272                    272                     ;MAKE N,V = 11
3887
3888  011160  002401                    2$:     BLT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3889  011162  000402                    BR      TST136          ;;GO TO NEXT TEST
3890
3891  011164  000000                    3$:     HALT            ;BLT FAILED
3892  011166  000772                    BR      1$              ;ERROR LOOP RETURN
3893
3894                                    ;;*****************************************************************
3895                                    ;*TEST 136      BASIC BGT TEST WITH Z = 1 AND N,V - 01
3896                                    ;;*****************************************************************
3897  011170                            TST136:
3898  011170  012700  000136            MOV     #136,R0         ;;LOAD R0 WITH TEST NUMBER
3899
3900  011174  000257                    1$:     CCC             ;CLEAR FLAGS
3901  011176  000266                    266                     ;SET Z AND V
3902
3903  011200  003001                    2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3904  011202  000402                    BR      TST137          ;;GO TO NEXT TEST
3905
3906  011204  000000                    3$:     HALT            ;BGT FAILED
3907  011206  000772                    BR      1$              ;ERROR LOOP RETURN
3908
3909                                    ;;*****************************************************************
3910                                    ;*TEST 137      BASIC BGT TEST WITH Z = 0 AND N,V = 01
3911                                    ;;*****************************************************************
3912  011210                            TST137:
3913  011210  012700  000137            MOV     #137,R0         ;;LOAD R0 WITH TEST NUMBER
3914
3915  011214  000257                    1$:     CCC             ;CLEAR FLAGS
3916  011216  000262                    SEV                     ;SET V
3917
```

```
3918  011220  003001              2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3919  011222  000402                      BR      TST140          ;;GO TO SCOPE LOOP EXIT
3920
3921  011224  000000              3$:     HALT                    ;BGT FAILED
3922  011226  000772                      BR      1$              ;ERROR LOOP RETURN
3923
3924                              ;;**********************************************************
3925                              ;*TEST 140       BASIC BGT TEST WITH Z = 1 AND N,V = 00
3926                              ;;**********************************************************
3927  011230                      TST140:
3928  011230  012700  000140              MOV     #140,R0         ;;LOAD R0 WITH TEST NUMBER
3929
3930  011234  000257              1$:     CCC                     ;CLEAR FLAGS
3931  011236  000264                      SEZ                     ;SET Z
3932
3933  011240  003001              2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3934  011242  000402                      BR      TST141          ;;GO TO SCOPE LOOP EXIT
3935
3936  011244  000000              3$:     HALT                    ;BGT FAILED
3937  011246  000772                      BR      1$              ;ERROR LOOP RETURN
3938
3939                              ;;**********************************************************
3940                              ;*TEST 141       BASIC BGT TEST WITH Z = 0 AND N,V = 00
3941                              ;;**********************************************************
3942  011250                      TST141:
3943  011250  012700  000141              MOV     #141,R0         ;;LOAD R0 WITH TEST NUMBER
3944
3945  011254  000257              1$:     CCC                     ;CLEAR FLAGS
3946
3947  011256                      2$:
3948  011256  003002                      BGT     TST142          ;;TEST THE BGT - IT SHOULD BR
3949
3950  011260  000000              3$:     HALT                    ;BGT FAILED
3951  011262  000774                      BR      1$              ;ERROR LOOP RETURN
3952
3953                              ;;**********************************************************
3954                              ;*TEST 142       BASIC BGT TEST WITH Z = 1 AND N,V = 01
3955                              ;;**********************************************************
3956  011264                      TST142:
3957  011264  012700  000142              MOV     #142,R0         ;;LOAD R0 WITH TEST NUMBER
3958
3959  011270  000257              1$:     CCC                     ;CLEAR FLAGS
3960  011272  000266                      266                     ;MAKE N,V = 01 AND Z = 1
3961
3962  011274  003001              2$:     BGT     3$              ;TEST THE BGT-IT SHOULDN'T BR
3963  011276  000402                      BR      TST143          ;;GO TO NEXT TEST
3964
3965  011300  000000              3$:     HALT                    ;BGT FAILED
3966  011302  000772                      BR      1$              ;ERROR LOOP RETURN
3967
3968                              ;;**********************************************************
3969                              ;*TEST 143       BASIC BGT TEST WITH Z = 1 AND N,V = 10
3970                              ;;**********************************************************
3971  011304                      TST143:
3972  011304  012700  000143              MOV     #143,R0         ;;LOAD R0 WITH TEST NUMBER
3973
```

```
3974  011310  000257                1$:     CCC                     ;CLEAR FLAGS
3975  011312  000274                        274                     ;MAKE Z = 1 AND N,V = 10
3976
3977  011314  003001                2$:     BGT     3$              ;TEST THE BLT-IT SHOULDN'T BR
3978  011316  000402                        BR      TST144          ;;GO TO NEXT TEST
3979
3980  011320  000000                3$:     HALT                    ;BLT FAILED
3981  011322  000772                        BR      1$              ;ERROR LOOP RETURN
3982
3983                                ;;*********************************************************
3984                                ;*TEST 144      BASIC BGT TEST WITH Z = 1 AND N,V = 11
3985                                ;;*********************************************************
3986  011324                        TST144:
3987  011324  012700  000144                MOV     #144,R0         ;;LOAD R0 WITH TEST NUMBER
3988
3989  011330  000257                1$:     CCC                     ;CLEAR FLAGS
3990  011332  000276                        276                     ;MAKE Z = 1 AND N,V = 11
3991
3992  011334  003001                2$:     BGT     3$              ;TEST THE BGT-IT SHOULD NOT BR
3993  011336  000402                        BR      TST145          ;;GO TO NEXT TEST
3994
3995  011340  000000                3$:     HALT                    ;BLT FAILED
3996  011342  000772                        BR      1$              ;ERROR LOOP RETURN
3997
3998                                ;;*********************************************************
3999                                ;*TEST 145      BASIC BGT TEST WITH Z=0 AND N,V=11
4000                                ;;*********************************************************
4001  011344                        TST145:
4002  011344  012700  000145                MOV     #145,R0         ;;LOAD R0 WITH TEST NUMBER
4003
4004  011350  000257                1$:     CCC                     ;CLEAR FLAGS
4005  011352  000272                        272                     ;MAKE N:C=1010
4006
4007  011354                        2$:
4008  011354  003002                        BGT     TST146          ;;TEST THE BGT - IT SHOULD BR
4009
4010  011356  000000                3$:     HALT                    ;BGT FAILED
4011  011360  000773                        BR      1$              ;ERROR LOOP RETURN
4012
4013                                ;;*********************************************************
4014                                ;*TEST 146      BASIC BHI TEST WITH Z,C = 00
4015                                ;;*********************************************************
4016  011362                        TST146:
4017  011362  012700  000146                MOV     #146,R0         ;;LOAD R0 WITH TEST NUMBER
4018
4019  011366  000257                1$:     CCC                     ;MAKE Z,C - 00
4020
4021  011370                        2$:
4022  011370  101002                        BHI     TST147          ;;TEST THE BHI-IT SHOULD BR
4023
4024  011372  000000                3$:     HALT                    ;BHI FAILED
4025  011374  000774                        BR      1$              ;ERROR LOOP RETURN
4026
4027                                ;;*********************************************************
4028                                ;*TEST 147      BASIC BHI TEST WITH Z,C = 01
4029                                ;;*********************************************************
```

```
 4030  011376
 4031  011376  012700  000147          TST147:  MOV     #147,R0         ;;LOAD R0 WITH TEST NUMBER
 4032
 4033  011402  000257                  1$:      CCC                     ;CLEAR FLAGS
 4034  011404  000261                           SEC                     ;MAKE Z,C = 01
 4035
 4036  011406  101001                  2$:      BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4037  011410  000402                           BR      TST150          ;;GO TO NEXT TEST
 4038
 4039  011412  000000                  3$:      HALT                    ;BHI FAILED
 4040  011414  000772                           BR      1$              ;ERROR LOOP RETURN
 4041
 4042                                  ;;*****************************************************************
 4043                                  ;*TEST 150      BASIC BHI TEST WITH Z,C = 10
 4044                                  ;;*****************************************************************
 4045  011416                          TST150:
 4046  011416  012700  000150                  MOV     #150,R0         ;;LOAD R0 WITH TEST NUMBER
 4047
 4048  011422  000257                  1$:      CCC                     ;CLEAR FLAGS
 4049  011424  000264                           SEZ                     ;MAKE Z,C = 10
 4050
 4051  011426  101001                  2$:      BHI     3$              ;TEST THE BHI-IT SHOULD NOT BR
 4052  011430  000402                           BR      TST151          ;;GO TO NEXT TEST
 4053
 4054  011432  000000                  3$:      HALT                    ;BHI FAILED
 4055  011434  000772                           BR      1$              ;ERROR LOOP RETURN
 4056
 4057                                  ;;*****************************************************************
 4058                                  ;*TEST 151      BASIC BHI TEST WITH Z,C = 11
 4059                                  ;;*****************************************************************
 4060  011436                          TST151:
 4061  011436  012700  000151                  MOV     #151,R0         ;;LOAD R0 WITH TEST NUMBER
 4062
 4063  011442  000257                  1$:      CCC                     ;CLEAR FLAGS
 4064  011444  000265                           265                     ;MAKE Z,C = 11
 4065
 4066  011446  101001                  2$:      BHI     3$              ;TEST THE BHI-IT SHOULDN'T BR
 4067  011450  000402                           BR      TST152          ;;GO TO NEXT TEST
 4068
 4069  011452  000000                  3$:      HALT                    ;BHI FAILED
 4070  011454  000772                           BR      1$              ;ERROR LOOP RETURN
 4071
 4072                                  ;;*****************************************************************
 4073                                  ;*TEST 152      BASIC NEG MODE 0 TEST : [DEST] GT 0
 4074                                  ;;*****************************************************************
 4075  011456                          TST152:
 4076  011456  012700  000152                  MOV     #152,R0         ;;LOAD R0 WITH TEST NUMBER
 4077  011462  012704  177776                  MOV     #-2,R4          ;RESULT S / B = 177776
 4078  011466  012703  000002          1$:     MOV     #2,R3           ;INITIAL [DEST] = 2
 4079  011472  000257                           CCC                     ;CLEAR FLAGS
 4080  011474  000266                           266                     ;MAKE N:C = 0110
 4081
 4082  011476  005403                  2$:      NEG     R3              ;TEST THE NEG
 4083
 4084  011500  100003                           BPL     3$
 4085  011502  001402                           BEQ     3$              ;DID N:C = 1001?
```

```
 4086  011504  102401                    BVS     3$
 4087  011506  103402                    BCS     4$
 4088
 4089  011510  000000              3$:   HALT                    ;NEGATE FAILED TO ALTER CODES PROPERLY
 4090  011512  000765                    BR      1$              ;ERROR LOOP RETURN
 4091
 4092  011514  020304              4$:   CMP     R3,R4           ;CORRECT RESULT?
 4093  011516  001402                    BEQ     TST153          ;;BR IF YES
 4094
 4095  011520  000000              5$:   HALT                    ;NEG DELIVERED WRONG RESULT
 4096  011522  000761                    BR      1$              ;ERROR LOOP RETURN
 4097
 4098                              ;;*****************************************************************
 4099                              ;*TEST 153      BASIC "SUB #,@#" TEST
 4100                              ;;*****************************************************************
 4101  011524                      TST153:
 4102  011524  012700  000153             MOV     #153,R0         ;;LOAD R0 WITH TEST NUMBER
 4103  011530  012704  000002             MOV     #2,R4           ;RESULT S / B = 2
 4104  011534  012702  063312             MOV     #MBUF0,R2       ;R2 POINTS TO DEST
 4105  011540  012712  000004       1$:   MOV     #4,(R2)         ;INITIAL [DEST] = 4
 4106  011544  000257                    CCC                     ;CLEAR FLAGS
 4107
 4108  011546  162737  000002  063312 2$:   SUB     #2,@#MBUF0      ;TEST THE SUB
 4109
 4110  011554  020412                    CMP     R4,(R2)         ;RESULT=2?
 4111  011556  001403                    BEQ     TST154          ;;BR IF YES
 4112  011560  011203                    MOV     (R2),R3         ;GET WAS DATA
 4113  011562  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4114  011564  000765                    BR      1$              ;ERROR LOOP RETURN
 4115
 4116                              ;;*****************************************************************
 4117                              ;*TEST 154      BASIC "SUB @#,RN" TEST
 4118                              ;;*****************************************************************
 4119  011566                      TST154:
 4120  011566  012700  000154             MOV     #154,R0         ;;LOAD R0 WITH TEST NUMBER
 4121  011572  012737  000002  063312     MOV     #2,@#MBUF0      ;SRC = 2
 4122  011600  012703  000004       1$:   MOV     #4,R3           ;INITIAL [DEST] = 4
 4123  011604  000257                    CCC                     ;SCOPE SYNC
 4124
 4125  011606  163703  063312       2$:   SUB     @#MBUF0,R3      ;TEST THE SUB
 4126
 4127  011612  020403                    CMP     R4,R3           ;RESULT=2?
 4128  011614  001402                    BEQ     TST155          ;;BR IF YES
 4129
 4130  011616  000000              3$:   HALT                    ;SUB DELIVERED WRONG RESULT
 4131  011620  000767                    BR      1$              ;ERROR LOOP RETURN
 4132
 4133                              ;;*****************************************************************
 4134                              ;*TEST 155      BASIC "RTS RN" TEST - <N:C> = 1111
 4135                              ;;*****************************************************************
 4136  011622                      TST155:
 4137  011622  012700  000155             MOV     #155,R0         ;;LOAD R0 WITH TEST NUMBER
 4138  011626  010605                    MOV     SP,R5           ;SAVE SP
 4139  011630  010506              1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4140  011632  012703  011652             MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM (R3)
 4141  011636  012746  177777             MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 FROM STACK
```

```
 4142  011642  000277                    SCC                     ;N:C = 1111
 4143
 4144  011644  000203            2$:     RTS     R3              ;TEST THE RTS - GO TO 4$
 4145
 4146  011646  000000            3$:     HALT                    ;RTS FAILED TO LOAD THE PC
 4147  011650  000767                    BR      1$              ;LOCK ON ERROR
 4148
 4149  011652  100003            4$:     BPL     5$              ;N:C = 1111 ?
 4150  011654  001002                    BNE     5$
 4151  011656  102001                    BVC     5$
 4152  011660  103402                    BCS     6$
 4153
 4154  011662  000000            5$:     HALT                    ;RTS ALTERED CODES - CLEARED ONE
 4155  011664  000761                    BR      1$              ;LOCK ON ERROR
 4156
 4157  011666  020327  177777    6$:     CMP     R3,#-1          ;DID R3 GET LOADED FROM STACK ?
 4158  011672  001402                    BEQ     8$              ;BR IF YES
 4159
 4160  011674  000000            7$:     HALT                    ;RTS FAILED TO LOAD REG
 4161  011676  000754                    BR      1$              ;LOCK ON ERROR
 4162
 4163  011700  020506            8$:     CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
 4164  011702  001402                    BEQ     TST156          ;;BR IF YES
 4165
 4166  011704  000000            9$:     HALT                    ;RTS FAILED TO POP SP
 4167  011706  000750                    BR      1$              ;LOCK ON ERROR
 4168
 4169                            ;;*******************************************************************
 4170                            ;*TEST 156      BASIC "RTS  PC" TEST
 4171                            ;;*******************************************************************
 4172  011710                    TST156:
 4173  011710  012700  000156           MOV     #156,R0         ;;LOAD R0 WITH TEST NUMBER
 4174  011714  010605                    MOV     SP,R5           ;SAVE THE ORIGINAL SP
 4175  011716  010506            1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4176  011720  012746  011734           MOV     #4$,-(SP)       ;PUSH NEW PC ON STACK
 4177  011724  000257                    CCC                     ;SCOPE SYNC
 4178
 4179  011726  000207            2$:     RTS     PC              ;TEST THE RTS - GO TO 4$
 4180
 4181  011730  000000            3$:     HALT                    ;RTS FAILED TO LOAD PC
 4182  011732  000771                    BR      1$              ;LOCK ON HARD ERROR
 4183
 4184  011734  020605            4$:     CMP     SP,R5           ;DID SP GET POPPED ?
 4185  011736  001402                    BEQ     TST157          ;;BR IF YES
 4186
 4187  011740  000000            5$:     HALT                    ;RTS FAILED TO UPDATE SP
 4188  011742  000765                    BR      1$              ;LOCK ON HARD ERROR
 4189
 4190                            ;;*******************************************************************
 4191                            ;*TEST 157      BASIC "JSR  PC,@#A" TEST
 4192                            ;;*******************************************************************
 4193  011744                    TST157:
 4194  011744  012700  000157           MOV     #157,R0         ;;LOAD R0 WITH TEST NUMBER
 4195                            .SBTTL USER CONTROLLED BREAKPOINT -- BIT3
 4196  011750  032737  000010  063234   BIT     #BIT3,@#BPTLOC  ;BREAKPOINT HALT SET ??
 4197  011756  001401                    BEQ     .+4             ;BR IF NOT
```

```
 4198  011760  000000                  HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 4199  011762  010605                  MOV     SP,R5           ;SAVE ORIGINAL SP
 4200  011764  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4201  011766  000257                  CCC                     ;SCOPE SYNC
 4202
 4203  011770  004737  012000  2$:     JSR     PC,@#4$ ;TEST THE JSR - GO TO 4$
 4204
 4205  011774  000000          3$:     HALT                    ;JSR FAILED TO LOAD PC
 4206  011776  000772                  BR      1$              ;LOCK ON HARD ERROR
 4207
 4208  012000  022726  011774  4$:     CMP     #3$,(SP)+       ;DID JSR SAVE OLD PC ON STACK ?
 4209  012004  001402                  BEQ     TST160          ;;BR IF YES
 4210
 4211  012006  000000          5$:     HALT                    ;JSR FAILED TO SAVE OLD PC
 4212  012010  000765                  BR      1$              ;LOCK ON HARD ERROR
 4213
 4214                          ;;*****************************************************************
 4215                          ;*TEST 160       BASIC "RTI" TEST - N:C=0000
 4216                          ;;*****************************************************************
 4217  012012                  TST160:
 4218  012012  012700  000160          MOV     #160,R0         ;;LOAD R0 WITH TEST NUMBER
 4219  012016  010605                  MOV     SP,R5           ;SAVE THE SP
 4220  012020  010506          1$:     MOV     R5,SP           ;RESET THE SP FOR ERROR LOOP
 4221  012022  012746  000357          MOV     #357,-(SP)      ;NEW PSW = 357
 4222  012026  012746  012046          MOV     #4$,-(SP)       ;NEW PC = 4$
 4223  012032  005037  177776          CLR     @#PSW           ;MAKE [PSW] = 000
 4224  012036  000257                  CCC                     ;MAKE N:C=0000
 4225
 4226  012040  000002          2$:     RTI                     ;TEST THE RTI - GO TO 4$
 4227
 4228  012042  000000          3$:     HALT                    ;RTI FAILED TO LOAD PC
 4229  012044  000765                  BR      1$              ;LOOP ON HARD ERROR
 4230
 4231  012046  013702  177776  4$:     MOV     @#PSW,R2        ;SAVE THE [PSW] IN R2
 4232  012052  022702  000357          CMP     #357,R2         ;WAS [PSW] = 357 ?
 4233  012056  001404                  BEQ     6$              ;BR IF YES
 4234
 4235  012060  010237  177776          MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
 4236  012064  000000          5$:     HALT                    ;RTI FAILED TO LOAD PSW
 4237  012066  000754                  BR      1$              ;LOCK ON HARD ERROR
 4238
 4239  012070  020605          6$:     CMP     SP,R5           ;DID SP GET UPDATED OK ?
 4240  012072  001402                  BEQ     TST161          ;;BR IF YES
 4241
 4242  012074  000000          7$:     HALT                    ;RTI FAILED TO UPDATE THE SP
 4243  012076  000750                  BR      1$              ;LOCK ON HARD ERROR
 4244
 4245                          ;;*****************************************************************
 4246                          ;*TEST 161       BASIC "RTI" TEST WITH N:C=1111
 4247                          ;;*****************************************************************
 4248  012100                  TST161:
 4249  012100  012700  000161          MOV     #161,R0         ;;LOAD R0 WITH TEST NUMBER
 4250  012104  010605                  MOV     SP,R5           ;SAVE THE SP IN R5
 4251  012106  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR _OOP
 4252  012110  005046                  CLR     -(SP)           ;NEW PSW - 000000
 4253  012112  012746  012130          MOV     #4$,-(SP)       ;NEW PC   4$
```

```
4254  012116  012737  000357  177776          MOV     #357,@#PSW      ;MAKE OLD PSW = 357
4255  012124  000240                          NOP                     ;SCOPE SYNC
4256
4257  012126  000002                  2$:     RTI                     ;TEST THE RTI - GO TO 4$
4258
4259  012130  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4260  012134  022702  000000                  CMP     #0,R2           ;WAS [PSW]=000
4261  012140  001404                          BEQ     TST162          ;;BR IF YES
4262
4263  012142  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4264  012146  000000                  3$:     HALT                    ;RTI FAILED TO CLEAR PSW
4265  012150  000756                          BR      1$              ;LOCK ON HARD ERROR
4266
4267                                  ;;*****************************************************************
4268                                  ;*TEST 162       BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
4269                                  ;;*****************************************************************
4270  012152                          TST162:
4271  012152  012700  000162                  MOV     #162,R0         ;;LOAD R0 WITH TEST NUMBER
4272  012156  010605                          MOV     SP,R5           ;SAVE THE SP
4273  012160  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
4274  012162  012737  012220  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
4275  012170  012737  000357  000022          MOV     #357,@#22
4276  012176  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
4277  012204  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
4278  012210  000257                          CCC                     ;SCOPE SYNC
4279
4280  012212  000004                  2$:     IOT                     ;TEST THE IOT
4281
4282  012214  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
4283  012216  000760                          BR      1$              ;LOCK ON HARD ERROR
4284
4285  012220  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
4286  012224  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
4287  012230  001404                          BEQ     6$              ;BR IF YES
4288
4289  012232  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4290  012236  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
4291  012240  000747                          BR      1$              ;LOCK ON HARD ERROR
4292
4293  012242  022726  012214          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
4294  012246  001404                          BEQ     8$              ;BR IF YES
4295
4296  012250  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4297  012254  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
4298  012256  000740                          BR      1$              ;LOCK ON HARD ERROR
4299
4300  012260  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
4301  012262  001404                          BEQ     TST163          ;;BR IF YES
4302
4303  012264  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
4304  012270  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
4305  012272  000732                          BR      1$              ;LOCK ON HARD ERROR
4306
4307                                  ;;*****************************************************************
4308                                  ;*TEST 163       BASIC "IOT" TEST - VERIFY LINKAGE TO SCOPE SERVICE
4309                                  ;;*****************************************************************
```

```
 4310  012274                          TST163:
 4311  012274  012700  000163                  MOV     #163,R0         ;;LOAD R0 WITH TEST NUMBER
 4312  012300  010605                          MOV     SP,R5           ;SAVE SP
 4313  012302  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4314  012304  005037  063244                  CLR     @#SCOFLG        ;TRAP SERVICE WILL COM "SCOFLG"
 4315  012310  012737  061612  000020          MOV     #SCOPEA,@#20    ;SET UP IOT VECTOR
 4316  012316  005037  000022                  CLR     @#22
 4317  012322  000257                          CCC                     ;SCOPE SYNC
 4318
 4319  012324  000004                  2$:     SCOPE                   ;TEST THE IOT
 4320
 4321  012326  005137  063244                  COM     @#SCOFLG        ;SCOFLG SHOULD BECOME 000000
 4322  012332  001402                          BEQ     4$              ;BR IF IT DID
 4323
 4324  012334  000000                  3$:     HALT                    ;IOT FAILED TO LINK TO SCOPE SERVICE
 4325  012336  000761                          BR      1$              ;LOCK ON HARD ERROR
 4326
 4327  012340  010506                  4$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
 4328                                  ;;*****************************************************************
 4329                                  ;*TEST 164       BASIC "IOT" TEST -VERIFY LOADING PSW WITH 357
 4330                                  ;;*****************************************************************
 4331  012342                          TST164:
 4332  012342  012700  000164                  MOV     #164,R0         ;;LOAD R0 WITH TEST NUMBER
 4333  012346  010605                          MOV     SP,R5           ;SAVE THE SP
 4334  012350  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4335  012352  012737  012410  000020          MOV     #4$,@#20        ;SET UP IOT VECTOR
 4336  012360  012737  000357  000022          MOV     #357,@#22
 4337  012366  012766  177777  177776          MOV     #-1,-2(SP)      ;IOT SHOULD CHANGE -1 TO 0
 4338  012374  005037  177776                  CLR     @#PSW           ;MAKE [PSW] = 000
 4339  012400  000257                          CCC                     ;SCOPE SYNC
 4340
 4341  012402  000004                  2$:     IOT                     ;TEST THE IOT
 4342
 4343  012404  000000                  3$:     HALT                    ;IOT FAILED TO LOAD PC
 4344  012406  000760                          BR      1$              ;LOCK ON HARD ERROR
 4345
 4346  012410  013702  177776          4$:     MOV     @#PSW,R2        ;GET THE PSW
 4347  012414  022702  000357                  CMP     #357,R2         ;DID IOT LOAD A 357 ?
 4348  012420  001404                          BEQ     6$              ;BR IF YES
 4349
 4350  012422  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4351  012426  000000                  5$:     HALT                    ;IOT FAILED TO LOAD PSW
 4352  012430  000747                          BR      1$              ;LOCK ON HARD ERROR
 4353
 4354  012432  022726  012404          6$:     CMP     #3$,(SP)+       ;DID IOT SAVE OLD PC ?
 4355  012436  001404                          BEQ     8$              ;BR IF YES
 4356
 4357  012440  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4358  012444  000000                  7$:     HALT                    ;IOT FAILED TO SAVE OLD PC
 4359  012446  000740                          BR      1$              ;LOCK ON HARD ERROR
 4360
 4361  012450  005726                  8$:     TST     (SP)+           ;DID IOT SAVE OLD PSW ?
 4362  012452  001404                          BEQ     ;ST165          ;;BR IF YES
 4363
 4364  012454  010237  177776                  MOV     R2,@#PSW        ;RESTORE ERROR PSW
 4365  012460  000000                  9$:     HALT                    ;IOT FAILED TO SAVE OLD PSW
```

```
                                        BR      1$              ;LOCK ON HARD ERROR

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

012516  000004          2$:     IOT                     ;TEST THE IOT

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

012526  010237  177776          MOV     R2,@#PSW        ;RESTORE THE ERROR PSW
012532  000000          3$:     HALT                    ;IOT FAILED TO CLEAR THE PSW
012534  000756                  BR      1$              ;LOCK ON HARD ERROR

012536  010506          6$:     MOV     R5,SP           ;RESET THE SP BEFORE CONTINUING

                        ;;*****************************************************************
                        ;*TEST 166       BASIC "TRAP" TEST - LINKAGE TO PRINT ROUTINE
                        ;;*****************************************************************
012540                  TST166:
012540  012700  000166          MOV     #166,R0         ;;LOAD R0 WITH TEST NUMBER
012544  010605                  MOV     SP,R5           ;SAVE THE SP
012546  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012550  005037  063236          CLR     @#PRIFLG        ;INITIALIZE TEST FLAG
012554  005037  000036          CLR     @#36            ;SET UP THE "TRAP" VECTOR
012560  012737  062202  000034  MOV     #PRINA,@#34
012566  000257                  CCC                     ;SCOPE SYNC

012570  104401          2$:     TYPE                    ;TEST THE TRAP

012572  012737  063166  000034  MOV     #$TRAP,@#34     ;SETUP TRAP VECTOR
012600  012737  000340  000036  MOV     #340,@#36
012606  005137  063236          COM     @#PRIFLG        ;SHOULD MAKE [PRIFLG] = 000000
012612  001402                  BEQ     TST167          ;;BR IF IT DID

012614  000000          3$:     HALT                    ;TRAP FAILED TO LINK TO PRINT SERV.
012616  000753                  BR      1$              ;LOCK ON HARD ERROR

                        ;;*****************************************************************
                        ;*TEST 167       BASIC "EMT" TEST - LINKAGE TO ERROR SERVICE
                        ;;*****************************************************************
012620                  TST167:
012620  012700  000167          MOV     #167,R0         ;;LOAD R0 WITH TEST NUMBER
012624  010605                  MOV     SP,R5           ;SAVE THE SP
012626  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
012630  012737  062040  000030  MOV     #ERRA,@#30      ;SET UP THE EMT VECTOR
012636  005037  000032          CLR     @#32
```

```
                                                                                                                                
 4422  012642  005037  063240                    CLR     @#ERRFLG        ;EMT SERVICE WILL COM [ERRFLG]
 4423  012646  000257                            CCC                     ;SCOPE SYNC
 4424
 4425  012650  104000                    2$:     ERROR                   ;TEST THE EMT
 4426
 4427  012652  005137  063240                    COM     @#ERRFLG        ;DID EMT SERV. COM ERRFLG?
 4428  012656  001402                            BEQ     TST170          ;;BR IF YES
 4429
 4430  012660  000000                    3$:     HALT                    ;EMT DID NOT LINK PROPERLY
 4431  012662  000761                            BR      1$              ;LOCK ON HARD ERROR
 4432                                    ;;**********************************************************************
 4433                                    ;*TEST 170      BASIC TEST OF RSVD INSTR. TRAP LINKAGE
 4434                                    ;;**********************************************************************
 4435  012664                            TST170:
 4436  012664  012700  000170                    MOV     #170,R0         ;;LOAD R0 WITH TEST NUMBER
 4437  012670  010605                            MOV     SP,R5           ;SAVE THE SP
 4438  012672  012737  061114  000010            MOV     #RSVTST,@#10    ;SET UP RSVD INSTR. TRAP VECTOR
 4439  012700  012737  000340  000012            MOV     #340,@#12
 4440  012706  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4441  012710  005037  063246                    CLR     @#RSVFLG        ;INITIALIZE TEST FLAG THAT WILL GET
 4442                                                                    ;COMPLEMENTED BY TRAP SERVICE
 4443  012714  000257                            CCC                     ;SCOPE SYNC
 4444
 4445  012716  000007                    2$:     000007                  ;FORCE RSVD INSTR. TRAP
 4446
 4447  012720  005137  063246                    COM     @#RSVFLG                ;TEST FLAG SHOULD GO TO 000000
 4448  012724  001402                            BEQ     4$              ;BR IF TRAP SPRUNG
 4449
 4450  012726  000000                    3$:     HALT                    ;RSVD INSTR. TRAP FAILED
 4451  012730  000766                            BR      1$              ;LOCK ON HARD ERROR
 4452
 4453  012732  012737  061122  000010    4$:     MOV     #RSERR,@#10     ;SET UP RSVD INSTR TRAP VECTOR TO POINT
 4454  012740  012737  000340  000012            MOV     #340,@#12       ;TO ERROR SERVICE ROUTINE
 4455
 4456                                    ;;**********************************************************************
 4457                                    ;*TEST 171      BASIC TEST OF BUS TIMEOUT TRAP LINKAGE
 4458                                    ;;**********************************************************************
 4459  012746                            TST171:
 4460  012746  012700  000171                    MOV     #171,R0         ;;LOAD R0 WITH TEST NUMBER
 4461  012752  010605                            MOV     SP,R5           ;SAVE THE SP
 4462  012754  012737  061212  000004            MOV     #BETST,@#4      ;SET UP THE BUS ERROR VECTOR
 4463  012762  012737  000340  000006            MOV     #340,@#6
 4464  012770  010506                    1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4465  012772  005037  063250                    CLR     @#BERFLG        ;INITIALIZE TEST FLAG THAT WILL GET
 4466                                                                    ;COMPLEMENTED BY TRAP SERVICE
 4467  012776  000257                            CCC                     ;SCOPE SYNC
 4468
 4469  013000  005737  177700            2$:     TST     @#177700        ;FORCE BUS TIMEOUT USING R0 ADDR.
 4470
 4471  013004  005137  063250                    COM     @#BERFLG        ;TEST FLAG SHOULD GO TO 000000
 4472  013010  001402                            BEQ     TST172          ;;BR IF TRAP SPRUNG
 4473
 4474  013012  000000                    3$:     HALT                    ;BUS ERROR FAILED TO SPRING TRAP
 4475  013014  000765                            BR      1$              ;LOCK ON HARD ERROR
 4476
 4477                                    ;;**********************************************************************
```

```
 4478                                   ;*TEST 172      BASIC TEST FOR ACCESSING DL11 REGISTERS
 4479                                   ;;**************************************************************
 4480  013016                           TST172:
 4481  013016  012700  000172                  MOV     #172,R0         ;;LOAD R0 WITH TEST NUMBER
 4482  013022  005067  050264                  CLR     MBUF0           ;INIT STALL COUNTER
 4483  013026  005367  050260           11$:   DEC     MBUF0           ;COUNT THE TIMER
 4484  013032  001375                          BNE     11$             ;BR IF NO TIMEOUT
 4485  013034  012737  013074  000004        MOV     #3$,@#4 ;SET UP BUS TIMEOUT VECTOR
 4486  013042  012737  000340  000006        MOV     #340,@#6
 4487  013050  010605                        MOV     SP,R5           ;SAVE TH SP
 4488  013052  010506                  1$:   MOV     R5,SP           ;RESET SP FOR ERROR LOOP
 4489  013054  012702  177560                MOV     #RCSR,R2        ;[R2] = STARTING DL11 ADDR.
 4490  013060  000257                        CCC                     ;SCOPE SYNC
 4491
 4492  013062  005722                  2$:   TST     (R2)+           ;REFERENCE DL11 - RCSR
 4493  013064  005722                        TST     (R2)+           ;REFERENCE DL11 - RDBR
 4494  013066  005722                        TST     (R2)+           ;REFERENCE DL11 - XCSR
 4495  013070  005712                        TST     (R2)            ;REFERENCE DL11 - XDBR
 4496
 4497  013072  000403                        BR      4$              ;GO TO NEXT TEST
 4498
 4499  013074  005742                  3$:   TST     -(R2)           ;BAD ADDRESS IN R2
 4500  013076  000000                        HALT                    ;ONE OF DL11 ADDR'S CAUSED TIME OUT
 4501  013100  000764                        BR      1$              ;LOCK ON HARD ERROR
 4502
 4503  013102  012737  061220  000004  4$:   MOV     #BERR,@#4       ;SET UP BUS ERROR VECTOR TO POINT
 4504  013110  012737  000340  000006        MOV     #340,@#6        ;TO ERROR SERVICE ROUTINE
 4505                                   ;;**************************************************************
 4506                                   ;*TEST 173      BASIC TEST OF DL11 - XCSR - READY(1)
 4507                                   ;;**************************************************************
 4508  013116                           TST173:
 4509  013116  012700  000173                  MOV     #173,R0         ;;LOAD R0 WITH TEST NUMBER
 4510  013122  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
 4511  013126  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
 4512  013132  005012                  1$:   CLR     (R2)            ;CLEAR [DEST]
 4513  013134  005001                        CLR     R1              ;SET UP TIMEOUT COUNTER
 4514  013136  000257                        CCC                     ;SCOPE SYNC
 4515
 4516  013140  020412                  2$:   CMP     R4,(R2)         ;TEST READY BIT - IT SHOULD BE SET
 4517
 4518  013142  001405                        BEQ     TST174          ;;BR IF IT WAS
 4519  013144  005301                        DEC     R1              ;TICK-TOCK GOES THE TIMER
 4520  013146  001374                        BNE     2$              ;BR IF NOT A TIMEOUT
 4521
 4522  013150  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 4523  013152  000000                  3$:   HALT                    ;READY BIT IN XCSR FAILED ON A (0)
 4524  013154  000766                        BR      1$              ;LOCK ON HARD ERROR
 4525
 4526                                   ;;**************************************************************
 4527                                   ;*TEST 174      BASIC TEST OF DL11 - XCSR - MAINT BIT (0)
 4528                                   ;;**************************************************************
 4529  013156                           TST174:
 4530  013156  012700  000174                  MOV     #174,R0         ;;LOAD R0 WITH TEST NUMBER
 4531  013162  012702  177564                  MOV     #XCSR,R2        ;DEST ADDR = XCSR
 4532  013166  012704  000200                  MOV     #200,R4         ;RESULT S / B = 200
 4533  013172  005012                  1$:   CLR     (R2)            ;CLEAR MAINT. BIT
```

```
4534  013174  000257                  CCC                     ;SCOPE SYNC
4535
4536  013176  020412          2$:     CMP     R4,(R2)         ;TEST MAINT(0)
4537
4538  013200  001403                  BEQ     TST175          ;;BR IF MAINT BIT CLEAR
4539
4540  013202  011203                  MOV     (R2),R3         ;GET THE WAS DATA
4541  013204  000000          3$:     HALT                    ;CAN'T CLEAR MAINT BIT
4542  013206  000771                  BR      1$              ;LOCK ON HARD ERROR
4543
4544                          ;;**********************************************************
4545                          ;*TEST 175      BASIC TEST OF DL11 XCSR - MAINT BIT - 1
4546                          ;;**********************************************************
4547  013210                  TST175:
4548  013210  012700  000175          MOV     #175,R0         ;;LOAD R0 WITH TEST NUMBER
4549  013214  012702  177564          MOV     #XCSR,R2        ;DEST ADDR = XCSR
4550  013220  012704  000204          MOV     #204,R4         ;RESULT S / B = 204
4551  013224  012712  000004  1$:     MOV     #4,(R2)         ;SET THE MAINT. BIT
4552  013230  000257                  CCC                     ;SCOPE SYNC
4553
4554  013232  020412          2$:     CMP     R4,(R2)         ;TEST MAINT.(1)
4555
4556  013234  001403                  BEQ     TST176          ;;BR IF IT WAS
4557
4558  013236  011203                  MOV     (R2),R3         ;GET THE WAS DATA
4559  013240  000000          3$:     HALT                    ;CAN'T SET MAINT BIT IN XCSR
4560  013242  000770                  BR      1$              ;LOCK 0 HARD ERROR
4561
4562                          ;;**********************************************************
4563                          ;*TEST 176      BASIC DL11 OUT / IN ECHO TEST (MAINT MODE)
4564                          ;THIS ROUTINE USES THE MAINTENANCE MODE FEATURE OF THE DL11 TO
4565                          ;TURN AROUND A STRING OF 8 CHARACTERS TO THE DL11. THIS STRING CONSISTS
4566                          ;OF ALTERNATING NULL / DELETE CHARS WHICH ARE NON PRINTING. THE 8 CHARS
4567                          ;ARE OUTPUT THEN READ BACK INTO A CORE BUFFER AND THEN THE INPUT AND
4568                          ;OUTPUT CORE BUFFERS ARE CHECKED FOR EQUIVALENCE. IF AN ERROR IS DET-
4569                          ;ECTED DURING THE COMPARISON THE ROUTINE HALTS WITH THE WAS AND S / B
4570                          ;DATA IN R3 AND R4 RESPECTFULLY. A TIMER IS EMPLOYED TO PREVENT THE
4571                          ;TEST FROM HANGING IF RECEIVER DONE DOES NOT RESPOND.
4572                          ;;**********************************************************
4573  013244                  TST176:
4574  013244  012700  000176          MOV     #176,R0         ;;LOAD R0 WITH TEST NUMBER
4575  013250  012702  177560  6$:     MOV     #RCSR,R2        ;R2 POINTS TO DL11 - START ADDR
4576  013254  105762  000002          TSTB    2(R2)           ;REFERENCE DL11 INPUT DATA BUFFER TWICE
4577  013260  105762  000002          TSTB    2(R2)           ;TO FLUSH RCVR "DONE" BIT
4578  013264  012703  063266          MOV     #IBUF,R3        ;R3 POINTS TO CORE INPUT BUFFER
4579  013270  012704  063256          MOV     #OBUF,R4        ;R4 POINTS TO CORE OUTPUT BUFFER
4580  013274  012705  000010          MOV     #10,R5          ;R5 WILL COUNT 8 CHARS OUTPUT
4581  013300  012762  000004  000004  MOV     #4,4(R2)        ;TURN ON MAINT MODE
4582
4583  013306  005001          1$:     CLR     R1              ;R1 USED AS TIMEOUT COUNTER
4584  013310  112462  000006          MOVB    (R4)+,6(R2)     ;LOAD OUTPUT BUFFER IN DL11
4585  013314  105712          2$:     TSTB    (R2)            ;RECEIVER DONE SET ?
4586  013316  100404                  BMI     3$              ;BR IF YES
4587  013320  005301                  DEC     R1              ;COUNT THE TIMER
4588  013322  001374                  BNE     2$              ;BR IF NO TIMEOUT
4589
```

```
 4590  013324  000000                    HALT                    ;DL11 FAILED TO RESPOND IN TIME
 4591  013326  000750                    BR      6$              ;LOCK ON HARD ERROR
 4592
 4593  013330  116223  000002    3$:     MOVB    2(R2),(R3)+     ;READ THE DL11 INPUT BUFFER INTO CORE
 4594  013334  005305                    DEC     R5              ;COUNT ONE CHAR
 4595  013336  001363                    BNE     1$              ;BR IF NOT DONE 8 CHARS
 4596
 4597  013340  005062  000004            CLR     4(R2)           ;TURN OFF MAINT. MODE
 4598  013344  012705  000010            MOV     #10,R5          ;RESET CHAR COUNTER
 4599  013350  012703  063266            MOV     #IBUF,R3        ;RESET INBUF POINTER
 4600  013354  012704  063256            MOV     #OBUF,R4        ;RESET OUTBUF POINTER
 4601
 4602  013360  122324            4$:     CMPB    (R3)+,(R4)+     ;INPUT = OUTPUT ??
 4603  013362  001003                    BNE     5$              ;BR IF NOT
 4604  013364  005305                    DEC     R5              ;COUNT ONE CHECKED
 4605  013366  001374                    BNE     4$              ;BR UNTIL 8 DONE
 4606  013370  000410                    BR      CITST           ;GO TO NEXT TEST
 4607
 4608  013372  114303            5$:     MOVB    -(R3),R3        ;WAS DATA IN R3 [BITS 7:0]
 4609  013374  114404                    MOVB    -(R4),R4        ;S / B DATA IN R4 [BITS 7:0]
 4610  013376  042703  177400            BIC     #177400,R3      ;STRIP OFF BITS <15:08>
 4611  013402  042704  177400            BIC     #177400,R4      ;  "             "
 4612  013406  000000                    HALT                    ;RECEIVED DATA NOT EQUAL TO OUTPUT DATA
 4613  013410  000717                    BR      6$              ;LOCK ON HARD ERROR
```

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

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

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

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

013576  104005          3$:     ERROR   5               ;BCC FAILED

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

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

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

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

013642  000277                SCC                     ;MAKE N:C = 1111

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

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

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

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

013702  000277                SCC                     ;MAKE N:C = 1111

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

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

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

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

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

013742  000257                CCC                     ;MAKE N:C - 0000

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

```
 4726  013746  013703  177776          MOV     @#PSW,R3        ;GET FLAGS
 4727  013752  005703                  TST     R3              ;N:C = 0000
 4728  013754  001401                  BEQ     TST204          ;;BR IF YES
 4729
 4730  013756  104001          3$:     ERROR   1               ;NO BRANCH MICROROUTINE-ALTERED CODES
 4731
 4732                          ;;*****************************************************************
 4733                          ;*TEST 204      VERIFY BRANCH MICROROUTINE DOES NOT SET FLAGS
 4734                          ;;*****************************************************************
 4735  013760                  TST204:
 4736  013760  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4737  013762  012700  000204          MOV     #204,R0         ;;LOAD R0 WITH TEST NUMBER
 4738  013766  013701  014002          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4739  013772  005004                  CLR     R4              ;PSW S/B = 0
 4740  013774  012702  177776          MOV     #PSW,R2         ;DEST = PSW FOR ERROR CALL
 4741
 4742  014000  000257                  CCC                     ;MAKE N:C = 0000
 4743
 4744  014002  000401          2$:     BR      4$              ;TEST THE BR
 4745
 4746  014004  104005          3$:     ERROR   5               ;JUST IN CASE THE BR DIDN'T WORK
 4747
 4748  014006  013703  177776  4$:     MOV     @#PSW,R3        ;GET FLAGS
 4749  014012  005703                  TST     R3              ;N:C = 0000
 4750  014014  001401                  BEQ     TST205          ;;BR IF YES
 4751
 4752  014016  104001          5$:     ERROR   1               ;BRANCH MICROROUTINE ALTERED CODES.
 4753
 4754                          ;;*****************************************************************
 4755                          ;*TEST 205      BLE TEST WITH Z = 0, AND N,V = 00
 4756                          ;;*****************************************************************
 4757  014020                  TST205:
 4758  014020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4759  014022  012700  000205          MOV     #205,R0         ;;LOAD R0 WITH TEST NUMBER
 4760  014026  013701  014034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4761  014032  000257                  CCC                     ;CLEAR FLAGS
 4762
 4763  014034  003401          2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
 4764  014036  000401                  BR      TST206          ;;GO TO SCOPE EXIT
 4765
 4766  014040  104005          3$:     ERROR   5               ;BLE FAILED
 4767
 4768                          ;;*****************************************************************
 4769                          ;*TEST 206      BLE TEST WITH Z = 1 AND N,V = 00
 4770                          ;;*****************************************************************
 4771  014042                  TST206:
 4772  014042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4773  014044  012700  000206          MOV     #206,R0         ;;LOAD R0 WITH TEST NUMBER
 4774  014050  013701  014060          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4775  014054  000257                  CCC                     ;CLEAR FLAGS
 4776  014056  000264                  SEZ                     ;SET Z = 1
 4777
 4778  014060                  2$:
 4779  014060  003401                  BLE     TST207          ;;TEST THE BLE-IT SHOULD BR
 4780
 4781  014062  104005          3$:     ERROR   5               ;BLE FAILED
```

```
                                ;;*****************************************************************
                                ;*TEST 207       BLE TEST WITH Z = 0 AND N,V = 01
                                ;;*****************************************************************
014064                          TST207:
014064  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014066  012700  000207                  MOV     #207,R0         ;;LOAD R0 WITH TEST NUMBER
014072  013701  014102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014076  000257                          CCC                     ;CLEAR FLAGS
014100  000262                          SEV                     ;MAKE Z = 0 AND N,V = 01

014102                          2$:
014102  003401                          BLE     TST210          ;;TEST THE BLE-IT SHOULD BR

014104  104005                  3$:     ERROR   5               ;BLE FAILED

                                ;;*****************************************************************
                                ;*TEST 210       BLE TEST WITH Z = 0 AND N,V = 10
                                ;;*****************************************************************
014106                          TST210:
014106  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014110  012700  000210                  MOV     #210,R0         ;;LOAD R0 WITH TEST NUMBER
014114  013701  014124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014120  000257                          CCC                     ;CLEAR FLAGS
014122  000270                          SEN                     ;MAKE Z = 0 AND N,V = 10

014124                          2$:
014124  003401                          BLE     TST211          ;;TEST THE BLE-IT SHOULD BR

014126  104005                  3$:     ERROR   5               ;BLE FAILED

                                ;;*****************************************************************
                                ;*TEST 211       BLE TEST WITH Z = 0 AND N,V = 11
                                ;;*****************************************************************
014130                          TST211:
014130  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014132  012700  000211                  MOV     #211,R0         ;;LOAD R0 WITH TEST NUMBER
014136  013701  014146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014142  000257                          CCC                     ;CLEAR FLAGS
014144  000272                          272                     ;MAKE Z = 0 AND N,V = 11

014146  003401                  2$:     BLE     3$              ;TEST THE BLE-IT SHOULDN'T BR
014150  000401                          BR      TST212          ;;GO TO SCOPE EXIT

014152  104005                  3$:     ERROR   5               ;BLE FAILED

                                ;;*****************************************************************
                                ;*TEST 212       BLOS TEST WITH Z,C = 00
                                ;;*****************************************************************
014154                          TST212:
014154  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014156  012700  000212                  MOV     #212,R0         ;;LOAD R0 WITH TEST NUMBER
014162  013701  014170                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014166  000257                          CCC .                   ;MAKE Z,C = 00

014170  101401                  2$:     BLOS    3$              ;TEST THE BLOS-IT SHOULDN'T BR
```

```
                                ;;*****************************************************************
                                ;*TEST 213       BLOS TEST WITH Z,C = 01
                                ;;*****************************************************************
014176                          TST213:
014176  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014200  012700  000213                  MOV     #213,R0         ;;LOAD R0 WITH TEST NUMBER
014204  013701  014214                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014210  000257                          CCC                     ;CLEAR FLAGS
014212  000261                          SEC                     ;MAKE Z,C = 01

014214                          2$:
014214  101401                          BLOS    TST214          ;;TEST THE BLOS-IT SHOULD BR

014216  104005                  3$:     ERROR   5               ;BLOS FAILED

                                ;;*****************************************************************
                                ;*TEST 214       BLOS TEST WITH Z,C = 10
                                ;;*****************************************************************
014220                          TST214:
014220  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014222  012700  000214                  MOV     #214,R0         ;;LOAD R0 WITH TEST NUMBER
014226  013701  014236                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014232  000257                          CCC                     ;CLEAR FLAGS
014234  000264                          SEZ                     ;MAKE Z,C = 10

014236                          2$:
014236  101401                          BLOS    TST215          ;;TEST THE BLOS-IT SHOULD BR

014240  104005                  3$:     ERROR   5               ;BLOS FAILED

                                ;;*****************************************************************
                                ;*TEST 215       BLOS TEST WITH Z,C = 11
                                ;;*****************************************************************
014242                          TST215:
014242  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014244  012700  000215                  MOV     #215,R0         ;;LOAD R0 WITH TEST NUMBER
014250  013701  014260                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
014254  000257                          CCC                     ;CLEAR FLAGS
014256  000265                          265                     ;MAKE Z,C - 11

014260                          2$:
014260  101401                          BLOS    TST216          ;;TEST THE BLOS-IT SHOULD BR

014262  104005                  3$:     ERROR   5               ;BLOS FAILED

                                ;;*****************************************************************
                                ;*TEST 216       SXT MODE 0 TEST WITH N = 0 AND C = 1
                                ;;*****************************************************************
014264                          TST216:
014264  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
014266  012700  000216                  MOV     #216,R0         ;;LOAD R0 WITH TEST NUMBER
014272  013701  014310                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 4894  014276  005004                          CLR     R4              ;RESULT S / B = 0
 4895  014300  012703  177777                  MOV     #-1,R3          ;INITAL DEST. OP = 177777
 4896  014304  000257                          CCC                     ;CLEAR CODES
 4897  014306  000263                          263                     ;N:C = 0011
 4898
 4899  014310  006703                  2$:     SXT     R3              ;TEST THE SXT
 4900
 4901  014312  100403                          BMI     3$
 4902  014314  001002                          BNE     3$              ;DID SXT MAKE N:C = 0101?
 4903  014316  102401                          BVS     3$
 4904  014320  103401                          BCS     4$
 4905
 4906  014322  104002                  3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
 4907
 4908  014324  005703                  4$:     TST     R3              ;DID RESULT = 0?
 4909  014326  001401                          BEQ     TST217          ;;BR IF IT DID
 4910
 4911  014330  104002                  5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT TO R3
 4912
 4913                                  ;;*******************************************************************
 4914                                  ;*TEST 217       SXT MODE 0 TEST WITH N = 0 AND C = 0
 4915                                  ;;*******************************************************************
 4916  014332                          TST217:
 4917  014332  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4918  014334  012700  000217                  MOV     #217,R0         ;;LOAD R0 WITH TEST NUMBER
 4919  014340  013701  014366                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4920                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT4
 4921  014344  032737  000020  063234          BIT     #BIT4,@#BPTLOC  ;BREAKPOINT HALT SET ??
 4922  014352  001401                          BEQ     .+4             ;BR IF NOT
 4923  014354  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 4924
 4925  014356  005004                          CLR     R4              ;RESULT S / B = 0
 4926  014360  012703  177777                  MOV     #-1,R3          ;INITIAL DEST OP = 177777
 4927  014364  000257                          CCC                     ;CLEAR N:C
 4928
 4929  014366  006703                  2$:     SXT     R3              ;TEST THE SXT
 4930  014370  103001                          BCC     TST220          ;;BR IF "C" STILL CLEAR
 4931
 4932  014372  104002                  3$:     ERROR   2               ;SXT AFFECTED "C" BIT
 4933
 4934                                  ;;*******************************************************************
 4935                                  ;*TEST 220       SXT MODE 0 TEST WITH N = 1 AND C = 1
 4936                                  ;;*******************************************************************
 4937  014374                          TST220:
 4938  014374  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 4939  014376  012700  000220                  MOV     #220,R0         ;;LOAD R0 WITH TEST NUMBER
 4940  014402  013701  014416                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 4941  014406  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
 4942  014412  005003                          CLR     R3              ;INITIAL DEST OP = 0
 4943  014414  000277                          SCC                     ;MAKE N:C = 1111
 4944
 4945  014416  006703                  2$:     SXT     R3              ;TEST THE SXT
 4946
 4947  014420  100003                          BPL     3$
 4948  014422  001402                          BEQ     3$              ;N:C - 1001?
 4949  014424  102401                          BVS     3$
```

```
4950  014426  103401                  BCS     4$
4951
4952  014430  104002          3$:     ERROR   2               ;SXT FAILED TO ALTER CODES PROPERLY
4953
4954  014432  010305          4$:     MOV     R3,R5           ;GET RESULT
4955  014434  005105                  COM     R5              ;COMPLEMENT IT-SHOULD GO TO 0
4956  014436  001401                  BEQ     TST221          ;;BR IF RESULT OF SXT = 1
4957
4958  014440  104002          5$:     ERROR   2               ;SXT DELIVERED WRONG RESULT.
4959
4960                          ;;*****************************************************************
4961                          ;*TEST 221       SXT MODE 0 TEST WITH N = 1 AND C = 0
4962                          ;;*****************************************************************
4963  014442                  TST221:
4964  014442  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4965  014444  012700  000221          MOV     #221,R0         ;;LOAD R0 WITH TEST NUMBER
4966  014450  013701  014466          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4967  014454  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
4968  014460  005003                  CLR     R3              ;INITIAL DEST OP = 0
4969  014462  000257                  CCC                     ;CLEAR FLAGS
4970  014464  000276                  276                     ;MAKE N:C = 1110
4971
4972  014466  006703          2$:     SXT     R3              ;TEST THE SXT
4973  014470  103001                  BCC     TST222          ;;BR IF "C" UNAFFECTED
4974
4975  014472  104002          3$:     ERROR   2               ;SXT SET "C" BIT
4976
4977                          ;;*****************************************************************
4978                          ;*TEST 222       SXT MODE 1 AND 2 TEST WITH N = 0 AND C = 1
4979                          ;;*****************************************************************
4980  014474                  TST222:
4981  014474  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
4982  014476  012700  000222          MOV     #222,R0         ;;LOAD R0 WITH TEST NUMBER
4983  014502  013701  014524          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
4984  014506  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
4985  014512  005004                  CLR     R4              ;RESULT S / B = 0
4986  014514  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
4987  014520  000257                  CCC                     ;CLEAR CODES
4988  014522  000263                  263                     ;MAKE N:C = 0011
4989
4990  014524  006712          2$:     SXT     (R2)            ;TEST THE SXT - DM1
4991
4992  014526  100403                  BMI     3$
4993  014530  001002                  BNE     3$              ;N:C - 0101
4994  014532  102401                  BVS     3$
4995  014534  103401                  BCS     4$
4996
4997  014536  104001          3$:     ERROR   1               ,SXT FAILED TO ALTER CODES PROPERLY
4998
4999  014540  005712          4$:     TST     (R2)            ;DID RESULT = 0?
5000  014542  001401                  BEQ     11$             ;BR IF YES
5001
5002  014544  104001          5$:     ERROR   1               ;SXT SHOULD HAVE ZEROED [DEST]
5003
5004  014546  012702  063312  11$:    MOV     #MBUF0,R2       ;DEST ADDR - MBUF0
5005  014552  013701  014566          MOV     @#12$,R1        ;LOAD R1 WITH TEST INSTR WORD
```

```
5006  014556  012712  177777          MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5007  014562  000257                   CCC                     ;CLEAR CODES
5008  014564  000263                   263                     ;MAKE N:C = 0011
5009
5010  014566  006722          12$:     SXT     (R2)+           ;TEST SXT - DM2
5011
5012  014570  100403                   BMI     7$              ;N:C = 0101 ?
5013  014572  001002                   BNE     7$
5014  014574  102401                   BVS     7$
5015  014576  103401                   BCS     6$
5016
5017  014600  104001          7$:      ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
5018
5019  014602  005737  063312    6$:    TST     @#MBUF0         ;DID RESULT GET ZEROED ?
5020  014606  001401                   BEQ     8$              ;BR IF YES
5021
5022  014610  104001          9$:      ERROR   1               ;SXT FAILED TO ZERO [DEST]
5023
5024  014612  020227  063314    8$:    CMP     R2,#MBUF0+2     ;WAS IT REALLY MODE 2 ?
5025  014616  001401                   BEQ     TST223          ;;BR IF YES
5026
5027  014620  104001                   ERROR   1               ;SXT FAILED TO AUTO INCREMEN~
5028
5029                            ;;*****************************************************************
5030                            ;*TEST 223       SXT MODE 1 TEST WITH N = 0 AND C = 0
5031                            ;;*****************************************************************
5032  014622                    TST223:
5033  014622  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5034  014624  012700  000223           MOV     #223,R0         ;;LOAD R0 WITH TEST NUMBER
5035  014630  013701  014650           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5036  014634  005004                   CLR     R4              ;RESULT S / B = 0
5037  014636  012702  063312           MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5038  014642  012712  177777           MOV     #-1,(R2)        ;INITIAL [DEST] = 177777
5039  014646  000257                   CCC                     ;CLEAR "C" BIT
5040
5041  014650  006712            2$.    SXT     (R2)            ;TEST THE SXT
5042  014652  103001                   BCC     TST224          ;;BR IF "C" UNDISTURBED
5043
5044  014654  104001            3$:    ERROR   1               ;SXT SET THE "C" BIT
5045
5046                            ;;*****************************************************************
5047                            ;*TEST 224       SXT MODE 1 TEST WITH N = 1 AND C = 1
5048                            ;;*****************************************************************
5049  014656                    TST224:
5050  014656  000004                   SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5051  014660  012700  000224           MOV     #224,R0         ;;LOAD R0 WITH TEST NUMBER
5052  014664  013701  014704           MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5053  014670  012704  177777           MOV     #-1,R4          ;RESULT S / B = 177777
5054  014674  012702  063312           MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5055  014700  005012                   CLR     (R2)            ;INITIAL [DEST] = 0
5056  014702  000277                   SCC                     ;MAKE N:C = 1111
5057
5058  014704  006712            2$:    SXT     (R2)            ;TEST THE SXT
5059
5060  014706  100003                   BPL     3$
5061  014710  001402                   BEQ     3$              ;N:C = 1001?
```

```
 5062  014712  102401                  BVS     3$
 5063  014714  103401                  BCS     4$
 5064
 5065  014716  104001            3$:   ERROR   1               ;SXT FAILED TO ALTER CODES PROPERLY
 5066
 5067  014720  021204            4$:   CMP     (R2),R4         ;RESULT = 177777?
 5068  014722  001401                  BEQ     TST225          ;;BR IF YES
 5069
 5070  014724  104001            5$:   ERROR   1               ;SXT DELIVERED WRONG RESULT
 5071
 5072                            ;;*****************************************************************
 5073                            ;*TEST 225      SXT MODE 1 TEST WITH N = 1 AND C = 0
 5074                            ;;*****************************************************************
 5075  014726                    TST225:
 5076  014726  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5077  014730  012700  000225          MOV     #225,R0         ;;LOAD R0 WITH TEST NUMBER
 5078  014734  013701  014756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5079  014740  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 5080  014744  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
 5081  014750  005012                  CLR     (R2)            ;INITIAL [DEST] = 0
 5082  014752  000257                  CCC                     ;CLEAR FLAGS
 5083  014754  000276                  276                     ;MAKE N:C = 1110
 5084
 5085  014756  006712            2$:   SXT     (R2)            ;TEST THE SXT
 5086  014760  103001                  BCC     TST226          ;;BR IF "C" UNAFFECTED
 5087
 5088  014762  104001            3$:   ERROR   1               ;SXT SET THE "C" BIT
 5089
 5090                            ;;*****************************************************************
 5091                            ;*TEST 226      SWAB MODE 0 TEST WITH POS. RESULT
 5092                            ;;*****************************************************************
 5093  014764                    TST226:
 5094  014764  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5095  014766  012700  000226          MOV     #226,R0         ;;LOAD R0 WITH TEST NUMBER
 5096  014772  013701  015012          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5097  014776  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
 5098  015002  012703  000377          MOV     #377,R3         ;INITIAL DEST OP = 377
 5099  015006  000257                  CCC                     ;CLEAR FLAGS
 5100  015010  000273                  273                     ;MAKE N:C - 1011
 5101
 5102  015012  000303            2$:   SWAB    R3              ;TEST THE SWAB
 5103
 5104  015014  100403                  BMI     3$
 5105  015016  001002                  BNE     3$              ;N·C = 0100
 5106  015020  102401                  BVS     3$
 5107  015022  103001                  BCC     4$
 5108
 5109  015024  104002            3$:   ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
 5110
 5111  015026  020403            4$:   CMP     R4,R3           ;CORRECT RESULT?
 5112  015030  001401                  BEQ     TST227          ;;BR IF YES
 5113
 5114  015032  104002            5$:   ERROR   2               ;SWAB DELIVERED WRONG RESULT
 5115
 5116                            ;;*****************************************************************
 5117                            ;*TEST 227      SWAB MODE 0 TEST WITH NEG. RESULT
```

```
5118                                    ;;*****************************************************************
5119  015034                            TST227:
5120  015034  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5121  015036  012700  000227                    MOV     #227,R0         ;;LOAD R0 WITH TEST NUMBER
5122  015042  013701  015062                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5123  015046  012704  000377                    MOV     #377,R4         ;RESULT S  /  B = 377
5124  015052  012703  177400                    MOV     #177400,R3      ;INITIAL DEST OP = 177400
5125  015056  000257                            CCC                     ;CLEAR FLAGS
5126  015060  000267                            267                     ;MAKE N:C - 0111
5127
5128  015062  000303                    2$:     SWAB    R3              ;TEST THE SWAB
5129
5130  015064  100003                            BPL     3$
5131  015066  001402                            BEQ     3$              ;DID SWAB MAKE N:C - 1000
5132  015070  102401                            BVS     3$
5133  015072  103001                            BCC     4$
5134
5135  015074  104002                    3$:     ERROR   2               ;SWAB FAILED TO ALTER CODES PROPERLY
5136
5137  015076  020403                    4$:     CMP     R4,R3           ;DID SWAB DELIVER CORRECT RESULT?
5138  015100  001401                            BEQ     TST230          ;;BR IF OK
5139
5140  015102  104002                    5$:     ERROR   2               ;SWAB DELIVERED WRONG RESULT
5141
5142                                    ;;*****************************************************************
5143                                    ;*TEST 230       SWAB MODE 1 AND 2 TEST WITH POS. RESULT
5144                                    ;;*****************************************************************
5145  015104                            TST230:
5146  015104  000004                            SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5147  015106  012700  000230                    MOV     #230,R0         ;;LOAD R0 WITH TEST NUMBER
5148  015112  013701  015136                    MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5149  015116  012704  177400                    MOV     #177400,R4      ;RESULT S  /  B = 177400
5150  015122  012702  063312                    MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5151  015126  012712  000377                    MOV     #377,(R2)       ;SET UP DEST OP = 377
5152  015132  000257                            CCC                     ;CLEAR FLAGS
5153  015134  000273                            273                     ;MAKE N:C = 1011
5154
5155  015136  000312                    2$:     SWAB    (R2)            ;TEST THE SWAB - DM1
5156
5157  015140  100403                            BMI     3$
5158  015142  001002                            BNE     3$              ;N:C = 0100
5159  015144  102401                            BVS     3$
5160  015146  103001                            BCC     4$
5161
5162  015150  104001                    3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5163
5164  015152  020412                    4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5165  015154  001401                            BEQ     5$              ;BR IF OK
5166
5167  015156  104001                            ERROR   1               ;SWAB DELIVERED WRONG RESULT
5168
5169  015160  013701  015200            5$:     MOV     @#20$,R1        ;LOAD R1 WITH TEST INSTR. WORD
5170  015164  012702  063312                    MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5171  015170  012712  000377                    MOV     #377,(R2)       ;[DEST] = 000377
5172  015174  000257                            CCC                     ;CLEAR FLAGS
5173  015176  000273                            273                     ;MAKE N:C = 1011
```

```
5174
5175  015200  000322          20$:    SWAB    (R2)+           ;TEST THE SWAB - DM2
5176
5177  015202  100403                  BMI     7$              ;N:C = 0100
5178  015204  001002                  BNE     7$
5179  015206  102401                  BVS     7$
5180  015210  103001                  BCC     6$
5181
5182  015212  104001          7$:     ERROR   1               ;SWAB FAILED TO SET CODES PROPERLY
5183
5184  015214  020437  063312  6$:     CMP     R4,@#MBUF0      ;CORRECT RESULT ?
5185  015220  001401                  BEQ     8$              ;BR IF YES
5186
5187  015222  104001          9$:     ERROR   1               ;SWAB DELIVERED THE WRONG RESULT
5188
5189  015224  020227  063314  8$:     CMP     R2,#MBUF0+2     ;DID AUTO INCREMENT OCCUR ?
5190  015230  001401                  BEQ     TST231          ;;BR IF YES
5191
5192  015232  104001                  ERROR   1               ;SWAB FAILED TO AUTO INC REG.
5193
5194                          ;;**********************************************************
5195                          ;*TEST 231       SWAB MODE 1 TEST WITH NEG. RESULT
5196                          ;;**********************************************************
5197  015234                  TST231:
5198  015234  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5199  015236  012700  000231          MOV     #231,R0         ;;LOAD R0 WITH TEST NUMBER
5200  015242  013701  015266          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5201  015246  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
5202  015252  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5203  015256  012712  177400          MOV     #177400,(R2)    ;SET UP DEST. OP = 177400
5204  015262  000257                  CCC                     ;CLEAR FLAGS
5205  015264  000267                  267                     ;MAKE N:C = 0111
5206
5207  015266  000312          2$:     SWAB    (R2)            ;TEST THE SWAB
5208
5209  015270  100003                  BPL     3$
5210  015272  001402                  BEQ     3$              ;N:C = 1000?
5211  015274  102401                  BVS     3$
5212  015276  103001                  BCC     4$
5213
5214  015300  104001          3$:     ERROR   1               ;SWAB FAILED TO ALTER CODES PROPERLY
5215
5216  015302  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT?
5217  015304  001401                  BEQ     TST232          ;;BR IF YES
5218
5219  015306  104001          5$:     ERROR   1               ;SWAB DELIVERED WRONG RESULT
5220
5221                          ;;**********************************************************
5222                          ;*TEST 232       NEG MODE 0 TEST : [DEST] = 0
5223                          ;;**********************************************************
5224  015310                  TST232:
5225  015310  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5226  015312  012700  000232          MOV     #232,R0         ;;LOAD R0 WITH TEST NUMBER
5227  015316  013701  015332          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5228  015322  005004                  CLR     R4              ;RESULT S / B = 0
5229  015324  005003                  CLR     R3              ;INITIAL [DEST] = 0
```

```
5230  015326  000257                  CCC                     ;CLEAR FLAGS
5231  015330  000273                  273                     ;MAKE N:C = 1011
5232
5233  015332  005403          2$:     NEG     R3              ;TEST THE NEG
5234
5235  015334  100403                  BMI     3$
5236  015336  001002                  BNE     3$              ;N:C = 0100 ONLY "Z" SET?
5237  015340  102401                  BVS     3$
5238  015342  103001                  BCC     4$
5239
5240  015344  104002          3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5241
5242  015346  020304          4$:     CMP     R3,R4           ;WAS RESULT = 0
5243  015350  001401                  BEQ     TST233          ;;BR IF YES
5244
5245  015352  104002          5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5246
5247                          ;;*******************************************************************
5248                          ;*TEST 233       NEG MODE 0 TEST : [DEST] LT 0
5249                          ;;*******************************************************************
5250  015354                  TST233:
5251  015354  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5252  015356  012700  000233          MOV     #233,R0         ;;LOAD R0 WITH TEST NUMBER
5253  015362  013701  015402          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5254  015366  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
5255  015372  012703  177776          MOV     #-2,R3          ;INITIAL [DEST] = 177776
5256  015376  000257                  CCC                     ;CLEAR FLAGS
5257  015400  000276                  276                     ;MAKE N:C = 1110
5258
5259  015402  005403          2$:     NEG     R3              ;TEST THE NEG
5260
5261  015404  100403                  BMI     3$
5262  015406  001402                  BEQ     3$              ;N:C = 0001?
5263  015410  102401                  BVS     3$
5264  015412  103401                  BCS     4$
5265
5266  015414  104002          3$:     ERROR   2               ;NEG FAILED TO ALTER CODES PROPERLY
5267
5268  015416  020304          4$:     CMP     R3,R4           ;RESULT = 2?
5269  015420  001401                  BEQ     TST234          ;;BR IF YES
5270
5271  015422  104002          5$:     ERROR   2               ;NEG DELIVERED WRONG RESULT
5272
5273                          ;;*******************************************************************
5274                          ;*TEST 234       NEG MODE 0 TEST : [DEST] = 100000 (8)
5275                          ;;*******************************************************************
5276  015424                  TST234:
5277  015424  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5278  015426  012700  000234          MOV     #234,R0         ;;LOAD R0 WITH TEST NUMBER
5279  015432  013701  015450          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5280  015436  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
5281  015442  010403                  MOV     R4,R3           ;INITIAL [DEST] = 100000
5282  015444  000257                  CCC                     ;CLEAR FLAGS
5283  015446  000264                  SEZ                     ;MAKE N:C = 01000
5284
5285  015450  005403          2$:     NEG     R3              ;TEST THE NEG
```

```
                                                                
 5286                                                           
 5287  015452  100003                BPL     3$                
 5288  015454  001402                BEQ     3$                ;N:C = 1011?
 5289  015456  102001                BVC     3$                
 5290  015460  103401                BCS     4$                
 5291                                                           
 5292  015462  104002          3$:   ERROR   2                 ;NEG FAILED TO ALTER CODES PROPERLY
 5293                                                           
 5294  015464  020304          4$:   CMP     R3,R4             ;RESULT STILL 100000?
 5295  015466  001401                BEQ     TST235            ;;BR IF YES
 5296                                                           
 5297  015470  104002          5$:   ERROR   2                 ;NEG DELIVERED WRONG RESULT
 5298                                                           
 5299                          ;;*******************************************************************
 5300                          ;*TEST 235       NEG MODE 1 TEST : [DEST] = 0
 5301                          ;;*******************************************************************
 5302  015472                TST235:                           
 5303  015472  000004                SCOPE                     ;CALL THE SCOPE LOOP UTILITY
 5304  015474  012700  000235        MOV     #235,R0           ;;LOAD R0 WITH TEST NUMBER
 5305  015500  013701  015520        MOV     @#2$,R1           ;LOAD R1 WITH TEST INSTRUCTION WORD
 5306  015504  012702  063312        MOV     #MBUF0,R2         ;R2 POINTS TO DEST OP
 5307  015510  005004                CLR     R4                ;RESULT S / B = 0
 5308  015512  005012                CLR     (R2)              ;INITIAL [DEST] = 0
 5309  015514  000257                CCC                       ;CLEAR FLAGS
 5310  015516  000273                273                       ;MAKE N:C = 1011
 5311                                                           
 5312  015520  005412          2$:   NEG     (R2)              ;TEST THE NEG
 5313                                                           
 5314  015522  100403                BMI     3$                
 5315  015524  001002                BNE     3$                ;N:C = 0100?
 5316  015526  102401                BVS     3$                
 5317  015530  103001                BCC     4$                
 5318                                                           
 5319  015532  104001          3$:   ERROR   1                 ;NEG FAILED TO ALTER CODES PROPERLY
 5320                                                           
 5321  015534  021204          4$:   CMP     (R2),R4           ;RESULT = 0?
 5322  015536  001401                BEQ     TST236            ;;BR IF YES
 5323                                                           
 5324  015540  104001          5$:   ERROR   1                 ;NEG DELIVERED WRONG RESULT
 5325                                                           
 5326                          ;;*******************************************************************
 5327                          ;*TEST 236       NEG MODE 1  TEST : [DEST] GT 0
 5328                          ;;*******************************************************************
 5329  015542                TST236:                           
 5330  015542  000004                SCOPE                     ;CALL THE SCOPE LOOP UTILITY
 5331  015544  012700  000236        MOV     #236,R0           ;;LOAD R0 WITH TEST NUMBER
 5332  015550  013701  015574        MOV     @#2$,R1           ;LOAD R1 WITH TEST INSTRUCTION WORD
 5333  015554  012702  063312        MOV     #MBUF0,R2         ;R2 POINTS TO DEST OP
 5334  015560  012704  177776        MOV     #-2,R4            ,RESULT S / B = 177776
 5335  015564  012712  000002        MOV     #2,(R2)           ;INITIAL [DEST] = 2
 5336  015570  000257                CCC                       ;CLEAR FLAGS
 5337  015572  000266                266                       ;MAKE N:C = 0110
 5338                                                           
 5339  015574  005412          2$:   NEG     (2)               ;TEST THE NEG
 5340                                                           
 5341  015576  100003                BPL     3$                
```

```
5342  015600  001402                  BEQ     3$              ;N:C = 1001?
5343  015602  102401                  BVS     3$
5344  015604  103401                  BCS     4$
5345
5346  015606  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5347
5348  015610  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT?
5349  015612  001401                  BEQ     TST237          ;;BR IF YES
5350
5351  015614  104001          5$:.    ERROR   1               ;NEG DELIVERED WRONG RESULT
5352
5353                          ;;**********************************************************************
5354                          ;*TEST 237       NEG MODE 1 TEST : [DEST] LT 0
5355                          ;;**********************************************************************
5356  015616                  TST237:
5357  015616  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5358  015620  012700  0C0237          MOV     #237,R0         ;;LOAD R0 WITH TEST NUMBER
5359  015624  013701  015650          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5360  015630  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5361  015634  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
5362  015640  012712  177776          MOV     #-2,(R2)        ;INITIAL [DEST] = 177776
5363  015644  000257                  CCC                     ;CLEAR FLAGS
5364  015646  000276                  276                     ;MAKE N:C = 1110
5365
5366  015650  005412          2$:     NEG     (R2)            ;TEST THE NEG
5367
5368  015652  100403                  BMI     3$
5369  015654  001402                  BEQ     3$              ;N:C = 0001?
5370  015656  102401                  BVS     3$
5371  015660  103401                  BCS     4$
5372
5373  015662  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5374
5375  015664  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT = 2?
5376  015666  001401                  BEQ     TST240          ;;BR IF YES
5377
5378  015670  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5379
5380                          ;;**********************************************************************
5381                          ;*TEST 240       NEG MODE 1 TEST: [DEST] = 100000 (8)
5382                          ;;**********************************************************************
5383  015672                  TST240:
5384  015672  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5385  015674  012700  000240          MOV     #240,R0         ;;LOAD R0 WITH TEST NUMBER
5386  015700  013701  015722          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5387  015704  012702  063312          MOV     #MBUF0,R2       ;R2 POINTS TO DEST OP
5388  015710  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
5389  015714  010412                  MOV     R4,(R2)         ;INITIAL [DEST] = 100000
5390  015716  000257                  CCC                     ;CLEAR FLAGS
5391  015720  000264                  SEZ                     ;MAKE N:Z = 0100
5392
5393  015722  005412          2$:     NEG     (R2)            ;TEST THE NEG
5394
5395  015724  100003                  BPL     3$
5396  015726  001402                  BEQ     3$              ;N:C = 1011?
5397  015730  102001                  BVC     3$
```

```
5398  015732  103401                  BCS     4$
5399
5400  015734  104001          3$:     ERROR   1               ;NEG FAILED TO ALTER CODES PROPERLY
5401
5402  015736  021204          4$:     CMP     (R2),R4         ;CORRECT RESULT = 100000?
5403  015740  001401                  BEQ     TST241          ;;BR IF YES
5404
5405  015742  104001          5$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
5406
5407                          ;;*******************************************************************
5408                          ;*TEST 241       ROR TEST - DM0 - N:C = 1110
5409                          ;;*******************************************************************
5410  015744                  TST241:
5411  015744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5412  015746  012700  000241          MOV     #241,R0         ;;LOAD R0 WITH TEST NUMBER
5413  015752  013701  015772          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5414  015756  012704  052525          MOV     #52525,R4       ;RESULT S / B = 52525
5415  015762  012703  125252          MOV     #125252,R3      ;[DEST] = 125252
5416  015766  000257                  CCC                     ;CLEAR FLAGS
5417  015770  000276                  276                     ;N:C = 1111
5418
5419  015772  006003          2$:     ROR     R3              ;TEST THE ROR
5420
5421  015774  100403                  BMI     3$              ;N:C = 0000 ?
5422  015776  001402                  BEQ     3$
5423  016000  102401                  BVS     3$
5424  016002  103001                  BCC     4$
5425
5426  016004  104002          3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5427
5428  016006  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
5429  016010  001401                  BEQ     TST242          ;;BR IF YES
5430
5431  016012  104002          5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5432
5433                          ;;*******************************************************************
5434                          ;*TEST 242       ROR TEST - DM0 - N:C = 1000
5435                          ;;*******************************************************************
5436  016014                  TST242:
5437  016014  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5438  016016  012700  000242          MOV     #242,R0         ;;LOAD R0 WITH TEST NUMBER
5439  016022  013701  016040          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5440  016026  005004                  CLR     R4              ;RESULT S / B = 000000
5441  016030  012703  000001          MOV     #1,R3           ;[DEST] = 1
5442  016034  000257                  CCC                     ;CLEAR FLAGS
5443  016036  000270                  SEN                     ;N:C = 1000
5444
5445  016040  006003          2$:     ROR     R3              ;TEST THE ROR
5446
5447  016042  100403                  BMI     3$              ;N:C = 0111 ?
5448  016044  001002                  BNE     3$
5449  016046  102001                  BVC     3$
5450  016050  103401                  BCS     4$
5451
5452  016052  104002          3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5453
```

```
5454  016054  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5455  016056  001401                          BEQ     TST243          ;;BR IF YES
5456
5457  016060  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5458
5459                                  ;;**********************************************************************
5460                                  ;*TEST 243       ROR TEST - DM0 - N:C = 0111
5461                                  ;;**********************************************************************
5462  016062                          TST243:
5463  016062  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5464  016064  012700  000243                  MOV     #243,R0         ;;LOAD R0 WITH TEST NUMBER
5465  016070  013701  016110                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5466  016074  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
5467  016100  012703  052525                  MOV     #52525,R3       ;[DEST] = 052525
5468  016104  000257                          CCC                     ;CLEAR FLAGS
5469  016106  000267                          267                     ;N:C = 0111
5470
5471  016110  006003                  2$:     ROR     R3              ;TEST THE ROR
5472
5473  016112  100003                          BPL     3$              ;N:C = 1001 ?
5474  016114  001402                          BEQ     3$
5475  016116  102401                          BVS     3$
5476  016120  103401                          BCS     4$
5477
5478  016122  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5479
5480  016124  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5481  016126  001401                          BEQ     TST244          ;;BR IF YES
5482
5483  016130  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
5484
5485                                  ;;**********************************************************************
5486                                  ;*TEST 244       ASR TEST - DM0 - N:C = 1000
5487                                  ;;**********************************************************************
5488  016132                          TST244:
5489  016132  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5490  016134  012700  000244                  MOV     #244,R0         ;;LOAD R0 WITH TEST NUMBER
5491  016140  013701  016156                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5492  016144  005004                          CLR     R4              ;RESULT S / B = 000000
5493  016146  012703  000001                  MOV     #1,R3           ;[DEST] = 1
5494  016152  000257                          CCC                     ;CLEAR FLAGS
5495  016154  000270                          SEN                     ;N:C = 1000
5496
5497  016156  006003                  2$:     ROR     R3              ;TEST THE ROR
5498
5499  016160  100403                          BMI     3$              ;N:C = 0111 ?
5500  016162  001002                          BNE     3$
5501  016164  102001                          BVC     3$
5502  016166  103401                          BCS     4$
5503
5504  016170  104002                  3$:     ERROR   2               ;ROR FAILED TO ALTER CODES PROPERLY
5505
5506  016172  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
5507  016174  001401                          BEQ     TST245          ;;BR IF YES
5508
5509  016176  104002                  5$:     ERROR   2               ;ROR DELIVERED THE WRONG RESULT
```

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

```
5566  016320                          TST247:
5567  016320  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5568  016322  012700  000247                  MOV     #247,R0         ;;LOAD R0 WITH TEST NUMBER
5569  016326  013701  016352                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5570  016332  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5571  016336  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
5572  016342  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
5573  016346  000257                          CCC                     ;CLEAR FLAGS
5574  016350  000276                          276                     ;N:C = 1110
5575
5576  016352  006012                  2$:     ROR     (R2)            ;TEST THE ROR
5577
5578  016354  100403                          BMI     3$              ;N·C - 0000 ?
5579  016356  001402                          BEQ     3$
5580  016360  102401                          BVS     3$
5581  016362  103001                          BCC     4$
5582
5583  016364  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5584
5585  016366  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5586  016370  001402                          BEQ     TST250          ;;BR IF YES
5587  016372  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5588  016374  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5589
5590                                  ;;**********************************************************
5591                                  ;*TEST 250       ROR TEST - DM1 - N:C = 1000
5592                                  ;;**********************************************************
5593  016376                          TST250:
5594  016376  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5595  016400  012700  000250                  MOV     #250,R0         ;;LOAD R0 WITH TEST NUMBER
5596  016404  013701  016426                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5597  016410  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5598  016414  005004                          CLR     R4              ;RESULT S / B = 000000
5599  016416  012712  000001                  MOV     #1,(R2)         ;[DEST] = 1
5600  016422  000257                          CCC                     ;CLEAR FLAGS
5601  016424  000270                          SEN                     ;N:C = 1000
5602
5603  016426  006012                  2$:     ROR     (R2)            ;TEST THE ROR
5604
5605  016430  100403                          BMI     3$              ;N:C = 0111 ?
5606  016432  001002                          BNE     3$
5607  016434  102001                          BVC     3$
5608  016436  103401                          BCS     4$
5609
5610  016440  104001                  3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5611
5612  016442  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5613  016444  001402                          BEQ     TST251          ;;BR IF YES
5614
5615  016446  011203                          MOV     (R2),R3         ;GET THE WAS DATA
5616  016450  104001                  5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5617
5618                                  ;;**********************************************************
5619                                  ;*TEST 251       ROR TEST - DM1 - N:C = 0111
5620                                  ;;**********************************************************
5621  016452                          TST251:
```

```
5622  016452  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5623  016454  012700  000251          MOV     #251,R0         ;;LOAD R0 WITH TEST NUMBER
5624  016460  013701  016504          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5625  016464  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5626  016470  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
5627  016474  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
5628  016500  000257                  CCC                     ;CLEAR FLAGS
5629  016502  000267                  267                     ;N:C = 0111
5630
5631  016504  006012          2$:     ROR     (R2)            ;TEST THE ROR
5632
5633  016506  100003                  BPL     3$              ;N:C = 1001 ?
5634  016510  001402                  BEQ     3$
5635  016512  102401                  BVS     3$
5636  016514  103401                  BCS     4$
5637
5638  016516  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5639
5640  016520  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5641  016522  001402                  BEQ     TST252          ;;BR IF YES
5642
5643  016524  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5644  016526  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5645
5646                          ;;**************************************************************
5647                          ;*TEST 252       ASR TEST - DM1 - N:C = 1000
5648                          ;;**************************************************************
5649  016530                  TST252:
5650  016530  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5651  016532  012700  000252          MOV     #252,R0         ;;LOAD R0 WITH TEST NUMBER
5652  016536  013701  016560          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5653  016542  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5654  016546  005004                  CLR     R4              ;RESULT S  /  B = 000000
5655  016550  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
5656  016554  000257                  CCC                     ;CLEAR FLAGS
5657  016556  000270                  SEN                     ;N:C = 1000
5658
5659  016560  006012          2$:     ROR     (R2)            ;TEST THE ROR
5660
5661  016562  100403                  BMI     3$              ;N:C = 0111 ?
5662  016564  001002                  BNE     3$
5663  016566  102001                  BVC     3$
5664  016570  103401                  BCS     4$
5665
5666  016572  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
5667
5668  016574  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5669  016576  001402                  BEQ     TST253          ;;BR IF YES
5670
5671  016600  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5672  016602  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
5673
5674                          ;;**************************************************************
5675                          ;*TEST 253       ASR TEST - DM1 - N:C = 1100
5676                          ;;**************************************************************
5677  016604                  TST253:
```

```
 5678  016604  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5679  016606  012700  000253          MOV     #253,R0         ;;LOAD R0 WITH TEST NUMBER
 5680  016612  013701  016636          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5681  016616  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5682  016622  012704  025252          MOV     #25252,R4       ;RESULT S  /  B = 25252
 5683  016626  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
 5684  016632  000257                  CCC                     ;CLEAR FLAGS
 5685  016634  000274                  274                     ;N:C = 1100
 5686
 5687  016636  006012          2$:     ROR     (R2)            ;TEST THE ROR
 5688
 5689  016640  100403                  BMI     3$              ;N:C = 0011 ?
 5690  016642  001402                  BEQ     3$
 5691  016644  102001                  BVC     3$
 5692  016646  103401                  BCS     4$
 5693
 5694  016650  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5695
 5696  016652  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5697  016654  001402                  BEQ     TST254          ;;BR IF YES
 5698
 5699  016656  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 5700  016660  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
 5701
 5702                          ;;**********************************************************
 5703                          ;*TEST 254      ASR TEST - DM1 - N:C = 0101
 5704                          ;;**********************************************************
 5705  016662                  TST254:
 5706  016662  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5707  016664  012700  000254          MOV     #254,R0         ;;LOAD R0 WITH TEST NUMBER
 5708  016670  013701  016714          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5709  016674  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5710  016700  012704  152525          MOV     #152525,R4      ;RESULT S / B = 152525
 5711  016704  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
 5712  016710  000257                  CCC                     ;CLEAR FLAGS
 5713  016712  000265                  265                     ;N:C = 0101
 5714
 5715  016714  006012          2$:     ROR     (R2)            ;TEST THE ROR
 5716
 5717  016716  100003                  BPL     3$              ;N:C = 1010 ?
 5718  016720  001402                  BEQ     3$
 5719  016722  102001                  BVC     3$
 5720  016724  103001                  BCC     4$
 5721
 5722  016726  104001          3$:     ERROR   1               ;ROR FAILED TO ALTER CODES PROPERLY
 5723
 5724  016730  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5725  016732  001402                  BEQ     TST255          ;;BR IF YES
 5726
 5727  016734  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 5728  016736  104001          5$:     ERROR   1               ;ROR DELIVERED WRONG RESULT
 5729
 5730                          ;;**********************************************************
 5731                          ;*TEST 255      RORB TEST - DM2 - EVEN ADDRESS
 5732                          ;;**********************************************************
 5733  016740                  TST255:
```

```
5734  016740  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5735  016742  012700  000255          MOV     #255,R0         ;;LOAD R0 WITH TEST NUMBER
5736  016746  013701  016772          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5737  016752  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5738  016756  012704  000177          MOV     #177,R4         ;RESULT S / B = 177
5739  016762  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5740  016764  012712  000377          MOV     #377,(R2)       ;[DEST] = 377
5741  016770  000257                  CCC                     ;SCOPE SYNC "C" = 0
5742
5743  016772  106023          2$:     RORB    (R3)+           ;TEST THE RORB
5744
5745  016774  103401                  BCS     4$              ;BR IF ROR SET "C"
5746
5747  016776  104001          3$:     ERROR   1               ;ROR FAILED TO SET "C"
5748
5749  017000  022703  063313  4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
5750  017004  001401                  BEQ     6$              ;BR IF YES
5751
5752  017006  104005          5$:     ERROR   5               ;RORB FAILED TO UPDATE DEST REG
5753
5754  017010  020412          6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5755  017012  001402                  BEQ     TST256          ;;BR IF YES
5756
5757  017014  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5758  017016  104001          7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5759
5760                          ;;**********************************************************
5761                          ;*TEST 256       RORB TEST - DM1 - EVEN ADDRESS
5762                          ;;**********************************************************
5763  017020                  TST256:
5764  017020  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
5765  017022  012700  000256          MOV     #256,R0         ;;LOAD R0 WITH TEST NUMBER
5766  017026  013701  017C54          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
5767  017032  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
5768  017036  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
5769  017042  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
5770  017044  012712  000376          MOV     #376,(R2)       ;[DEST] = 376
5771  017050  000257                  CCC                     ;CLEAR FLAGS
5772  017052  000261                  SEC                     ;SCOPE SYNC - SET "C"
5773
5774  017054  106013          2$:     RORB    (R3)            ;TEST THE RORB
5775
5776  017056  103001                  BCC     4$              ;BR IF "C" CLR - IT SHOULD BE
5777
5778  017060  104001          3$:     ERROR   1               ;RORB FAILED TO CLR "C"
5779
5780  017062  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
5781  017064  001402                  BEQ     TST257          ;;BR IF YES
5782
5783  017066  011203                  MOV     (R2),R3         ;GET THE WAS DATA
5784  017070  104001          5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
5785
5786                          ;;**********************************************************
5787                          ;*TEST 257       RORB TEST - DM2 - ODD ADDRESS
5788                          ;;**********************************************************
5789  017072                  TST257:
```

```
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #257,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT5
                                        BIT     #BIT5,@#BPTLOC  ;BREAKPOINT HALT SET ??
                                        BEQ     .+4             ;BR IF NOT
                                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
                                        MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
                                        MOV     #77777,R4       ;RESULT S / B = 77777
                                        MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
                                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
                                        MOV     #-1,(R5)        ;[DEST] = 177777
                                        CLC                     ;SCOPE SYNC - "C" =0

                                2$:     RORB    (R3)+           ;TEST THE RORB

                                        BCS     4$              ;BR IF "C" IS SET - IT SHOULD BE

                                3$:     ERROR   1               ;RORB FAILED TO SET "C"

                                4$:     CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
                                        BEQ     6$              ;BR IF YES

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

                                6$:     CMP     R4,(R5)         ;CORRECT RESULT ?
                                        BEQ     TST260          ;;BR IF YES

                                        MOV     (R5),R3         ;GET THE WAS DATA
                                7$:     ERROR   1               ;RORB DELIVERED WRONG RESULT

                                ;;**************************************************************
                                ;*TEST 260      RORB TEST - DM1 - ODD ADDRESS
                                ;;**************************************************************
                                TST260:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #260,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
                                        MOV     #-1,R4          ;RESULT S / B = 177777
                                        MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
                                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
                                        MOV     #177377,(R5)    ;[DEST] = 177377
                                        SEC                     ;SCOPE SYNC - SET "C"

                                2$:     RORB    (R3)+           ;TEST THE RORB

                                        BCC     4$              ;BR IF "C" CLEAR - IT SHOULD BE

                                3$:     ERROR   1               ;RORB FAILED TO CLEAR "C"

                                4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
                                        BEQ     TST261          ;;BR IF YES

                                        MOV     (R5),R3         ;GET THE WAS DATA
                                5$:     ERROR   1               ;RORB DELIVERED WRONG RESULT
```

```
                                ;;*****************************************************************
                                ;*TEST 261       ASRB TEST - DM2 - ODD ADDRESS
                                ;;*****************************************************************
017244                          TST261:
017244  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017246  012700  000261                  MOV     #261,R0         ;;LOAD R0 WITH TEST NUMBER
017252  013701  017302                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017256  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
017262  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
017266  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
017272  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017274  012715  000777                  MOV     #777,(R5)       ;[DEST] = 777
017300  000257                          CCC                     ;SCOPE SYNC "C" = 0

017302  106223          2$:     ASRB    (R3)+                   ;TEST THE ASRB

017304  103401                  BCS     4$                      ;BR IF CARRY SET - IT SHOULD BE

017306  104001          3$:     ERROR   1                       ;ASRB FAILED TO SET THE CARRY

017310  022703  063314  4$:     CMP     #MBUF0+2,R3             ;DID DEST REG GET INCREMENTED ?
017314  001401                  BEQ     6$                      ;BR IF YES

017316  104005          5$:     ERROR   5                       ;ASRB FAILED TO UPDATE DEST REG

017320  020415          6$:     CMP     R4,(R5)                 ;CORRECT RESULT ?
017322  001402                  BEQ     TST262                  ;;BR IF YES

017324  011503                  MOV     (R5),R3                 ;GET THE WAS DATA
017326  104001          7$:     ERROR   1                       ;ASRB DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 262       ASRB TEST - DM1 - ODD ADDRESS
                                ;;*****************************************************************
017330                          TST262:
017330  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017332  012700  000262                  MOV     #262,R0         ;;LOAD R0 WITH TEST NUMBER
017336  013701  017366                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017342  012702  063313                  MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
017346  012704  140377                  MOV     #140377,R4      ;RESULT S / B = 140377
017352  012705  063312                  MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
017356  010203                          MOV     R2,R3           ;R3 CONTAINS DEST ADDR
017360  012715  100377                  MOV     #100377,(R5)    ;[DEST] = 100377
017364  000261                          SEC                     ;SCOPE SYNC - "C" = 1

017366  106213          2$:     ASRB    (R3)                    ;TEST THE ASRB

017370  103001                  BCC     4$                      ;BR IF CARRY CLEAR - IT SHOULD BE

017372  104001          3$:     ERROR   1                       ;ASRB FAILED TO CLEAR THE CARRY

017374  020415          4$:     CMP     R4,(R5)                 ;CORRECT RESULT ?
017376  001402                  BEQ     TST263                  ;;BR IF YES

017400  011503                  MOV     (R5),R3                 ;GET THE WAS DATA
```

```
 5902  017402  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
 5903
 5904                                  ;;*********************************************************************
 5905                                  ;*TEST 263       ASRB TEST - DM2 - EVEN ADDRESS
 5906                                  ;;*********************************************************************
 5907  017404                          TST263:
 5908  017404  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5909  017406  012700  000263                 MOV     #263,R0         ;;LOAD R0 WITH TEST NUMBER
 5910  017412  013701  017436                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5911  017416  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5912  017422  012704  000077                 MOV     #77,R4          ;RESULT S / B = 77
 5913  017426  010203                         MOV     R2,R3           ;R3 CONTAINS DEST ADDR
 5914  017430  012712  000177                 MOV     #177,(R2)       ;[DEST] = 177
 5915  017434  000257                         CCC                     ;SCOPE SYNC - "C" = 0
 5916
 5917  017436  106223                  2$:     ASRB    (R3)+           ;TEST THE ASRB
 5918
 5919  017440  103401                          BCS     4$              ;BR IF "C" = 1 - IT SHOULD BE
 5920
 5921  017442  104001                  3$:     ERROR   1               ;ASRB FAILED TO SET "C"
 5922
 5923  017444  022703  063313          4$:     CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
 5924  017450  001401                          BEQ     6$              ;BR IF YES
 5925
 5926  017452  104005                  5$:     ERROR   5               ;ASRB FAILED TO UPDATE DEST REG
 5927
 5928  017454  020412                  6$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5929  017456  001402                          BEQ     TST264          ;;BR IF YES
 5930
 5931  017460  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 5932  017462  104001                  7$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
 5933
 5934                                  ;;*********************************************************************
 5935                                  ;*TEST 264       ASRB TEST - DM1 - EVEN ADDRESS
 5936                                  ;;*********************************************************************
 5937  017464                          TST264:
 5938  017464  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 5939  017466  012700  000264                 MOV     #264,R0         ;;LOAD R0 WITH TEST NUMBER
 5940  017472  013701  017516                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 5941  017476  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 5942  017502  012704  000303                 MOV     #303,R4         ;RESULT S / B = 303
 5943  017506  010203                         MOV     R2,R3           ;R3 CONTAINS DEST ADDR
 5944  017510  012712  000206                 MOV     #206,(R2)       ;[DEST] - 206
 5945  017514  000261                         SEC                     ;SCOPE SYNC - "C" - 1
 5946
 5947  017516  106213                  2$:     ASRB    (R3)            ;TEST THE CLRASRB
 5948
 5949  017520  103001                          BCC     4$              ;BR IF CARRY CLEAR - ITSHOULD BE
 5950
 5951  017522  104001                  3$:     ERROR   1               ;ASRB FAILED TO CLEAR THE CARRY
 5952
 5953  017524  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 5954  017526  001402                          BEQ     TST265          ;;BR IF YES
 5955
 5956  017530  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 5957  017532  104001                  5$:     ERROR   1               ;ASRB DELIVERED WRONG RESULT
```

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

017556  005703          2$:             TST     R3              ;TEST THE TST

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

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

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

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

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

017624  005703          2$:             TST     R3              ;TEST THE TST

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

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

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

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

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

```
                                ;;*****************************************************************
017646                          TST267:
017646  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017650  012700  000267                  MOV     #267,R0         ;;LOAD R0 WITH TEST NUMBER
017654  013701  017672                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017660  005004                          CLR     R4              ;RESULT S / B = 000000
017662  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
017666  000257                          CCC                     ;CLEAR CODES
017670  000273                          273                     ;N:C = 1011

017672  005003                  2$:     CLR     R3              ;TEST THE CLR

017674  100403                          BMI     3$              ;N:C = 0100 ?
017676  001002                          BNE     3$
017700  102401                          BVS     3$
017702  103001                          BCC     4$

017704  104002                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY

017706  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017710  001401                          BEQ     TST270          ;;BR IF YES

017712  104002                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 270       CLR DMO TEST - N:C = 0000
                                ;;*****************************************************************
017714                          TST270:
017714  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017716  012700  000270                  MOV     #270,R0         ;;LOAD R0 WITH TEST NUMBER
017722  013701  017736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
017726  005004                          CLR     R4              ;RESULT S / B = 000000
017730  012703  177777                  MOV     #-1,R3          ;[DEST] = 177777
017734  000257                          CCC                     ;CLEAR CODES

017736  005003                  2$:     CLR     R3              ;TEST THE CLR

017740  100403                          BMI     3$              ;N:C = 0100 ?
017742  001002                          BNE     3$
017744  102401                          BVS     3$
017746  103001                          BCC     4$

017750  104002                  3$:     ERROR   2               ;CLR FAILED TO ALTER THE CODES PROPERLY

017752  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
017754  001401                          BEQ     TST271          ;;BR IF YES

017756  104002                  5$:     ERROR   2               ;CLR DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 271       COM DMO TEST - N:C = 0110
                                ;;*****************************************************************
017760                          TST271:
017760  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
017762  012700  000271                  MOV     #271,R0         ;;LOAD R0 WITH TEST NUMBER
017766  013701  020006                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 6070  017772  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
 6071  017776  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
 6072  020002  000257                  CCC                     ;CLEAR CODES
 6073  020004  000266                  266                     ;N:C = 0110
 6074
 6075  020006  005103          2$:     COM     R3              ;TEST THE COM
 6076
 6077  020010  100003                  BPL     3$              ;N:C = 1001 ?
 6078  020012  001402                  BEQ     3$
 6079  020014  102401                  BVS     3$
 6080  020016  103401                  BCS     4$
 6081
 6082  020020  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
 6083
 6084  020022  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6085  020024  001401                  BEQ     TST272          ;;BR IF YES
 6086
 6087  020026  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
 6088
 6089                          ;;*******************************************************************
 6090                          ;*TEST 272       COM DMO TEST - N:C = 1001
 6091                          ;;*******************************************************************
 6092  020030                  TST272:
 6093  020030  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6094  020032  012700  000272          MOV     #272,R0         ;;LOAD R0 WITH TEST NUMBER
 6095  020036  013701  020054          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6096  020042  005004                  CLR     R4              ;RESULT S / B = 000000
 6097  020044  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
 6098  020050  000257                  CCC                     ;CLEAR CODES
 6099  020052  000271                  271                     ;N:C = 1001
 6100
 6101  020054  005103          2$:     COM     R3              ;TEST THE COM
 6102
 6103  020056  100403                  BMI     3$              ;N:C = 0101 ?
 6104  020060  001002                  BNE     3$
 6105  020062  102401                  BVS     3$
 6106  020064  103401                  BCS     4$
 6107
 6108  020066  104002          3$:     ERROR   2               ;COM FAILED TO ALTER THE CODES PROPERLY
 6109
 6110  020070  020403          4$:     CMP     R4,R3           ;RESULT OK ?
 6111  020072  001401                  BEQ     TST273          ;;BR IF YES
 6112
 6113  020074  104002          5$:     ERROR   2               ;COM DELIVERED THE WRONG RESULT
 6114
 6115                          ;;*******************************************************************
 6116                          ;*TEST 273       INC DMO TEST - N:C = 1011
 6117                          ;;*******************************************************************
 6118  020076                  TST273:
 6119  020076  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6120  020100  012700  000273          MOV     #273,R0         ;;LOAD R0 WITH TEST NUMBER
 6121  020104  013701  020122          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6122  020110  005004                  CLR     R4              ;RESULT S / B = 000000
 6123  020112  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
 6124  020116  000257                  CCC                     ;CLEAR CODES
 6125  020120  000273                  273                     ;N:C = 1011
```

```
                                ;TEST THE INC
                                ;N:C = 0101 ?
                                ;INC FAILED TO ALTER THE CODES PROPERLY
                                ;RESULT OK ?
                                ;;BR IF YES
                                ;INC DELIVERED THE WRONG RESULT
                ;;**************************************************************
                ;*TEST 274       INC DMO TEST - N:C = 0100
                ;;**************************************************************
                TST274:
                                ;CALL THE SCOPE LOOP UTILITY
                        MOV     #274,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        MOV     #100000,R4      ;RESULT S / B = 100000
                        MOV     #77777,R3       ;[DEST] = 77777
                        CCC                     ;CLEAR CODES
                        264                     ;N:C = 0100

                2$:     INC     R3              ;TEST THE INC

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

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

                4$:     CMP     R4,R3           ;RESULT OK ?
                        BEQ     TST275          ;;BR IF YES

                5$:     ERROR   2               ;INC DELIVERED THE WRONG RESULT

                ;;**************************************************************
                ;*TEST 275       DEC DMO TEST - N:C = 1011
                ;;**************************************************************
                TST275:
                                ;CALL THE SCOPE LOOP UTILITY
                        MOV     #275,R0         ;;LOAD R0 WITH TEST NUMBER
                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                        CLR     R4              ;RESULT S / B = 000000
                        MOV     #1,R3           ;[DEST] = 1
                        CCC                     ;CLEAR CODES
                        273                     ;N:C = 1011

                2$:     DEC     R3              ;TEST THE DEC

                        BMI     3$              ;N:C = 0101 ?
```

```
 6182  020244  001002                    BNE     3$
 6183  020246  102401                    BVS     3$
 6184  020250  103401                    BCS     4$
 6185
 6186  020252  104002            3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6187
 6188  020254  020403            4$:     CMP     R4,R3           ;RESULT OK ?
 6189  020256  001401                    BEQ     TST276          ;;BR IF YES
 6190
 6191  020260  104002            5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6192
 6193                            ;;*****************************************************************
 6194                            ;*TEST 276       DEC DMO TEST - N:C = 1100
 6195                            ;;*****************************************************************
 6196  020262                    TST276:
 6197  020262  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6198  020264  012700  000276            MOV     #276,R0         ;;LOAD R0 WITH TEST NUMBER
 6199  020270  013701  020310            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6200  020274  012704  077777            MOV     #77777,R4       ;RESULT S / B = 77777
 6201  020300  012703  100000            MOV     #100000,R3      ;[DEST] = 100000
 6202  020304  000257                    CCC                     ;CLEAR CODES
 6203  020306  000274                    274                     ;N:C = 1100
 6204
 6205  020310  005303            2$:     DEC     R3              ;TEST THE DEC
 6206
 6207  020312  100403                    BMI     3$              ;N:C = 0010 ?
 6208  020314  001402                    BEQ     3$
 6209  020316  102001                    BVC     3$
 6210  020320  103001                    BCC     4$
 6211
 6212  020322  104002            3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6213
 6214  020324  020403            4$:     CMP     R4,R3           ;RESULT OK ?
 6215  020326  001401                    BEQ     TST277          ;;BR IF YES
 6216
 6217  020330  104002            5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT
 6218
 6219                            ;;*****************************************************************
 6220                            ;*TEST 277       DEC DMO TEST - N:C = 0000
 6221                            ;;*****************************************************************
 6222  020332                    TST277:
 6223  020332  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6224  020334  012700  000277            MOV     #277,R0         ;;LOAD R0 WITH TEST NUMBER
 6225  020340  013701  020354            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6226  020344  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
 6227  020350  005003                    CLR     R3              ;[DEST] = 000000
 6228  020352  000257                    CCC                     ;CLEAR CODES
 6229
 6230  020354  005303            2$:     DEC     R3              ;TEST THE DEC
```

```
                                                                ;N:C = 1000 ?
020356  100003                        BPL     3$
020360  001402                        BEQ     3$
020362  102401                        BVS     3$
020364  103001                        BCC     4$

020366  104002                3$:     ERROR   2               ;DEC FAILED TO ALTER THE CODES PROPERLY

020370  020403                4$:     CMP     R4,R3           ;RESULT OK ?
020372  001401                        BEQ     TST300          ;;BR IF YES

020374  104002                5$:     ERROR   2               ;DEC DELIVERED THE WRONG RESULT

                              ;;*****************************************************************
                              ;*TEST 300       ASL DMO TEST - N:C = 1000
                              ;;*****************************************************************
020376                        TST300:
020376  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020400  012700  000300                MOV     #300,R0         ;;LOAD R0 WITH TEST NUMBER
020404  013701  020422                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020410  005004                        CLR     R4              ;RESULT S / B = 000000
020412  012703  100000                MOV     #100000,R3      ;[DEST] = 100000
020416  000257                        CCC                     ;CLEAR CODES
020420  000270                        SEN                     ;N:C = 1000

020422  006303                2$:     ASL     R3              ;TEST THE ASL

020424  100403                        BMI     3$              ;N:C = 0111 ?
020426  001002                        BNE     3$
020430  102001                        BVC     3$
020432  103401                        BCS     4$

020434  104002                3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY

020436  020403                4$:     CMP     R4,R3           ;RESULT OK ?
020440  001401                        BEQ     TST301          ;;BR IF YES

020442  104002                5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT

                              ;;*****************************************************************
                              ;*TEST 301       ASL DMO TEST - N:C = 0101
                              ;;*****************************************************************
020444                        TST301:
020444  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020446  012700  000301                MOV     #301,R0         ;;LOAD R0 WITH TEST NUMBER
020452  013701  020472                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020456  012704  100000                MOV     #100000,R4      ;RESULT S / B = 100000
020462  012703  040000                MOV     #40000,R3       ;[DEST] = 40000
020466  000257                        CCC                     ;CLEAR CODES
020470  000265                        265                     ;N:C = 0101

020472  006303                2$:     ASL     R3              ;TEST THE ASL

020474  100003                        BPL     3$              ;N:C = 1010 ?
020476  001402                        BEQ     3$
020500  102001                        BVC     3$
```

```
  6287  020502  103001                  BCC     4$
  6288
  6289  020504  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
  6290
  6291  020506  020403          4$:     CMP     R4,R3           ;RESULT OK ?
  6292  020510  001401                  BEQ     TST302          ;;BR IF YES
  6293
  6294  020512  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
  6295
  6296                          ;;*********************************************************
  6297                          ;*TEST 302       ASL DMO TEST - N:C = 0010
  6298                          ;;*********************************************************
  6299  020514                  TST302:
  6300  020514  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
  6301  020516  012700  000302          MOV     #302,R0         ;;LOAD R0 WITH TEST NUMBER
  6302  020522  013701  020536          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
  6303  020526  005004                  CLR     R4              ;RESULT S  /  B = 000000
  6304  020530  005003                  CLR     R3              ;[DEST] = 000000
  6305  020532  000257                  CCC                     ;CLEAR CODES
  6306  020534  000262                  SEV                     ;N:C = 0010
  6307
  6308  020536  006303          2$:     ASL     R3              ;TEST THE ASL
  6309
  6310  020540  100403                  BMI     3$              ;N:C = 0100 ?
  6311  020542  001002                  BNE     3$
  6312  020544  102401                  BVS     3$
  6313  020546  103001                  BCC     4$
  6314
  6315  020550  104002          3$:     ERROR   2               ;ASL FAILED TO ALTER THE CODES PROPERLY
  6316
  6317  020552  020403          4$:     CMP     R4,R3           ;RESULT OK ?
  6318  020554  001401                  BEQ     TST303          ;;BR IF YES
  6319
  6320  020556  104002          5$:     ERROR   2               ;ASL DELIVERED THE WRONG RESULT
  6321
  6322                          ;;*********************************************************
  6323                          ;*TEST 303       ROL DMO TEST - N:C = 1101
  6324                          ;;*********************************************************
  6325  020560                  TST303:
  6326  020560  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
  6327  020562  012700  000303          MOV     #303,R0         ;;LOAD R0 WITH TEST NUMBER
  6328  020566  013701  020606          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
  6329  020572  012704  052525          MOV     #52525,R4       ;RESULT S  /  B - 52525
  6330  020576  012703  125252          MOV     #125252,R3      ;[DEST] = 125252
  6331  020602  000257                  CCC                     ;CLEAR CODES
  6332  020604  000275                  275                     ;N:C = 1101
  6333
  6334  020606  006103          2$:     ROL     R3              ;TEST THE ROL
  6335
  6336  020610  100403                  BMI     3$              ;N:C - 0011 ?
  6337  020612  001402                  BEQ     3$
  6338  020614  102001                  BVC     3$
  6339  020616  103401                  BCS     4$
  6340
  6341  020620  104002          3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
  6342
```

```
                                ;;*****************************************************************
                                ;*TEST 304       ROL DMO TEST - N:C = 0101
                                ;;*****************************************************************
020630                          TST304:
020630  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020632  012700  000304                  MOV     #304,R0         ;;LOAD R0 WITH TEST NUMBER
020636  013701  020656                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020642  012704  125253                  MOV     #125253,R4      ;RESULT S / B = 125253
020646  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
020652  000257                          CCC                     ;CLEAR CODES
020654  000265                          265                     ;N:C = 0101

020656  006103                  2$:     ROL     R3              ;TEST THE ROL

020660  100003                          BPL     3$              ;N:C = 1010 ?
020662  001402                          BEQ     3$
020664  102001                          BVC     3$
020666  103001                          BCC     4$

020670  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY
020672  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
020674  001401                          BEQ     TST305          ;;BR IF YES

020676  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 305       ROL DMO TEST - N:C = 0010
                                ;;*****************************************************************
020700                          TST305:
020700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
020702  012700  000305                  MOV     #305,R0         ;;LOAD R0 WITH TEST NUMBER
020706  013701  020722                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
020712  005004                          CLR     R4              ;RESULT S / B = 000000
020714  005003                          CLR     R3              ;[DEST] = 000000
020716  000257                          CCC                     ;CLEAR CODES
020720  000262                          SEV                     ;N:C = 0010

020722  006103                  2$:     ROL     R3              ;TEST THE ROL

020724  100403                          BMI     3$              ;N:C = 0100 ?
020726  001002                          BNE     3$
020730  102401                          BVS     3$
020732  103001                          BCC     4$

020734  104002                  3$:     ERROR   2               ;ROL FAILED TO ALTER THE CODES PROPERLY

020736  020403                  4$:     CMP     R4,R3           ;RESULT OK ?
020740  001401                          BEQ     TST306          ;;BR IF YES

020742  104002                  5$:     ERROR   2               ;ROL DELIVERED THE WRONG RESULT
```

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

020772  005503                  2$:     ADC     R3              ;TEST THE ADC

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

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

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

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

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

021040  005503                  2$:     ADC     R3              ;TEST THE ADC

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

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

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

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

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

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

                                2$:     ADC     R3              ;TEST THE ADC

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

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

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

                                5$:     ERROR   2               ;ADC DELIVERED THE WRONG RESULT

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

                                2$:     SBC     R3              ;TEST THE SBC

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

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

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

                                5$:     ERROR   2               ;SBC DFLIVERED THE WRONG RESULT

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

```
 6511  021216  012703  100000                MOV     #100000,R3      ;[DEST] = 100000
 6512  021222  000257                        CCC                     ;CLEAR CODES
 6513  021224  000265                        265                     ;N:C = 0101
 6514
 6515  021226  005603              2$:       SBC     R3              ;TEST THE SBC
 6516
 6517  021230  100403                        BMI     3$              ;N:C = 0010 ?
 6518  021232  001402                        BEQ     3$
 6519  021234  102001                        BVC     3$
 6520  021236  103001                        BCC     4$
 6521
 6522  021240  104002              3$:       ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6523
 6524  021242  020403              4$:       CMP     R4,R3           ;RESULT OK ?
 6525  021244  001401                        BEQ     TST313          ;;BR IF YES
 6526
 6527  021246  104002              5$:       ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6528
 6529                              ;;*********************************************************
 6530                              ;*TEST 313       SBC DMO TEST - N:C = 1110
 6531                              ;;*********************************************************
 6532  021250                      TST313:
 6533  021250  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6534  021252  012700  000313                MOV     #313,R0         ;;LOAD R0 WITH TEST NUMBER
 6535  021256  013701  021276                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6536  021262  012704  000001                MOV     #1,R4           ;RESULT S / B = 1
 6537  021266  012703  000001                MOV     #1,R3           ;[DEST] = 1
 6538  021272  000257                        CCC                     ;CLEAR CODES
 6539  021274  000276                        276                     ;N:C = 1110
 6540
 6541  021276  005603              2$:       SBC     R3              ;TEST THE SBC
 6542
 6543  021300  100403                        BMI     3$              ;N:C = 0000 ?
 6544  021302  001402                        BEQ     3$
 6545  021304  10240*                        BVS     3$
 6546  021306  103001                        BCC     4$
 6547
 6548  021310  104002              3$:       ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6549
 6550  021312  020403              4$:       CMP     R4,R3           ;RESULT OK ?
 6551  021314  001401                        BEQ     TST314          ;;BR IF YES
 6552
 6553  021316  104002              5$:       ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6554
 6555                              ;;*********************************************************
 6556                              ;*TEST 314       SBC DMO TEST - N:C = 0111
 6557                              ;;*********************************************************
 6558  021320                      TST314:
 6559  021320  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6560  021322  012700  000314                MOV     #314,R0         ;;LOAD R0 WITH TEST NUMBER
 6561  021326  013701  021344                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6562  021332  012704  177777                MOV     #-1,R4          ;RESULT S / B = 177777
 6563  021336  005003                        CLR     R3              ;[DEST] = 000000
 6564  021340  000257                        CCC                     ;CLEAR CODES
 6565  021342  000267                        267                     ;N:C = 0111
 6566
```

```
 6567  021344  005603                2$:     SBC     R3              ;TEST THE SBC
 6568
 6569  021346  100003                        BPL     3$              ;N:C = 1001 ?
 6570  021350  001402                        BEQ     ?$
 6571  021352  102401                        BVS     3$
 6572  021354  103401                        BCS     4$
 6573
 6574  021356  104002                3$:     ERROR   2               ;SBC FAILED TO ALTER THE CODES PROPERLY
 6575
 6576  021360  020403                4$:     CMP     R4,R3           ;RESULT OK ?
 6577  021362  001401                        BEQ     TST315          ;;BR IF YES
 6578
 6579  021364  104002                5$:     ERROR   2               ;SBC DELIVERED THE WRONG RESULT
 6580
 6581                                ;;*********************************************************
 6582                                ;*TEST 315       TST DM1 TEST - N:C = 1011
 6583                                ;;*********************************************************
 6584  021366                        TST315:
 6585  021366  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6586  021370  012700  000315                MOV     #315,R0         ;;LOAD R0 WITH TEST NUMBER
 6587  021374  013701  021414                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6588  021400  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6589  021404  005004                        CLR     R4              ;RESULT S / B = 000000
 6590  021406  005012                        CLR     (R2)            ;[DEST] = 000000
 6591  021410  000257                        CCC                     ;CLEAR CODES
 6592  021412  000273                        273                     ;N:C=1011
 6593
 6594  021414  005712                2$:     TST     (R2)            ;TEST THE TST
 6595
 6596  021416  100403                        BMI     3$              ;N:C = 0100 ?
 6597  021420  001002                        BNE     3$
 6598  021422  102401                        BVS     3$
 6599  021424  103001                        BCC     4$
 6600
 6601  021426  104001                3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
 6602
 6603  021430  020412                4$:     CMP     R4,(R2)         ;RESULT OK ?
 6604  021432  001402                        BEQ     TST316          ;;BR IF YES
 6605
 6606  021434  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6607  021436  104001                5$:     ERROR   1               ;TST ALTERED THE [DEST]
 6608
 6609                                ;;*********************************************************
 6610                                ;*TEST 316       TST DM1 TEST - N:C = 0100
 6611                                ;;*********************************************************
 6612  021440                        TST316:
 6613  021440  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6614  021442  012700  000316                MOV     #316,R0         ;;LOAD R0 WITH TEST NUMBER
 6615  021446  013701  021472                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6616  021452  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6617  021456  005004                        CLR     R4
 6618  021460  005104                        COM     R4              ;RESULT S / B = 177777
 6619  021462  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
 6620  021466  000257                        CCC                     ;CLEAR CODES
 6621  021470  000264                        264                     ;N:C=0100
 6622
```

```
                                                    1 10
 6623  021472  005712                 2$:     TST     (R2)            ;TEST THE TST
 6624
 6625  021474  100003                         BPL     3$              ;N:C = 1000 ?
 6626  021476  001402                         BEQ     3$
 6627  021500  102401                         BVS     3$
 6628  021502  103001                         BCC     4$
 6629
 6630  021504  104001                 3$:     ERROR   1               ;TST FAILED TO ALTER CODES PROPERLY
 6631
 6632  021506  020412                 4$:     CMP     R4,(R2)         ;RESULT OK ?
 6633  021510  001402                         BEQ     TST317          ;;BR IF YES
 6634
 6635  021512  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 6636  021514  104001                 5$:     ERROR   1               ;TST ALTERED THE [DEST]
 6637
 6638                                 ;;**************************************************************
 6639                                 ;*TEST 317      CLR DM1 TEST - N:C = 1011
 6640                                 ;;**************************************************************
 6641  021516                         TST317:
 6642  021516  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6643  021520  012700  000317                 MOV     #317,R0         ;;LOAD R0 WITH TEST NUMBER
 6644  021524  013701  021560                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6645                                 .SBTTL USER CONTROLLED BREAKPOINT -- BIT6
 6646  021530  032737  000100  063234         BIT     #BIT6,@#BPTLOC  ;BREAKPOINT HALT SET ??
 6647  021536  001401                         BEQ     .+4             ;BR IF NOT
 6648  021540  000000                         HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 6649  021542  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6650  021546  005004                         CLR     R4              ;RESULT S / B = 000000
 6651  021550  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
 6652  021554  000257                         CCC                     ;CLEAR CODES
 6653  021556  000273                         273                     ;N:C = 1011
 6654
 6655  021560  005012                 2$:     CLR     (R2)            ;TEST THE CLR
 6656
 6657  021562  100403                         BMI     3$              ;N:C = 0100 ?
 6658  021564  001002                         BNE     3$
 6659  021566  102401                         BVS     3$
 6660  021570  103001                         BCC     4$
 6661
 6662  021572  104001                 3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
 6663
 6664  021574  020412                 4$:     CMP     R4,(R2)         ;RESULT OK ?
 6665  021576  001402                         BEQ     TST320          ;;BR IF YES
 6666
 6667  021600  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 6668  021602  104001                 5$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
 6669
 6670                                 ;;**************************************************************
 6671                                 ;*TEST 320      CLR DM2 TEST - N:C = 0000
 6672                                 ;;**************************************************************
 6673  021604                         TST320:
 6674  021604  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6675  021606  012700  000320                 MOV     #320,R0         ;;LOAD R0 WITH TEST NUMBER
 6676  021612  013701  021632                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6677  021616  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6678  021622  005004                         CLR     R4              ;RESULT S / B = 000000
```

```
6679  021624  013712  063324          MOV     @#DWTA+2,(R2)   ;[DEST] = 177777
6680  021630  000257                  CCC                     ;CLEAR CODES
6681
6682  021632  005022          2$:     CLR     (R2)+           ;TEST THE CLR
6683
6684  021634  100403                  BMI     3$              ;N:C = 0100 ?
6685  021636  001002                  BNE     3$
6686  021640  102401                  BVS     3$
6687  021642  103001                  BCC     4$
6688
6689  021644  104001          3$:     ERROR   1               ;CLR FAILED TO ALTER THE CODES PROPERLY
6690
6691  021646  022702  063314  4$:     CMP     #MBUF0+2,R2     ;DID CLR INCREMENT DEST REG
6692  021652  001401                  BEQ     6$              ;BR IF YES
6693
6694  021654  104005          5$:     ERROR   5               ;CLR FAILED TO UPDATE DEST REG
6695
6696  021656  020442          6$:     CMP     R4,-(R2)                ;RESULT OK ?
6697  021660  001402                  BEQ     TST321          ;;BR IF YES
6698
6699  021662  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6700  021664  104001          7$:     ERROR   1               ;CLR DELIVERED THE WRONG RESULT
6701
6702                          ;;*****************************************************************
6703                          ;*TEST 321      COM DM1 TEST - N:C   0110
6704                          ;;*****************************************************************
6705  021666                  TST321:
6706  021666  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6707  021670  012700  000321          MOV     #321,R0         ;;LOAD R0 WITH TEST NUMBER
6708  021674  013701  021720          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
6709  021700  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
6710  021704  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
6711  021710  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
6712  021714  000257                  CCC                     ;CLEAR CODES
6713  021716  000266                  266                     ;N:C = 0110
6714
6715  021720  005112          2$:     COM     (R2)            ;TEST THE CLR
6716
6717  021722  100003                  BPL     3$              ;N:C = 1001 ?
6718  021724  001402                  BEQ     3$
6719  021726  102401                  BVS     3$
6720  021730  103401                  BCS     4$
6721
6722  021732  104001          3$:     ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
6723  021734  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
6724  021736  001402                  BEQ     TST322          ;;BR IF YES
6725
6726  021740  011203                  MOV     (R2),R3         ;GET THE WAS DATA
6727  021742  104001          5$:     ERROR   1               ;COM DELIVERED THE WRONG RESULT
6728
6729                          ;;*****************************************************************
6730                          ;*TEST 322      COM DM1 TEST - N:C = 1001
6731                          ;;*****************************************************************
6732  021744                  TST322:
6733  021744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
6734  021746  012700  000322          MOV     #322,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 6735  021752  013701  021774                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6736  021756  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6737  021762  005004                        CLR     R4              ;RESULT S / B = 000000
 6738  021764  012712  177777                MOV     #-1,(R2)        ;[DEST] = 177777
 6739  021770  000257                        CCC                     ;CLEAR CODES
 6740  021772  000271                        271                     ;N:C = 1001
 6741
 6742  021774  005112             2$:        COM     (R2)            ;TEST THE COM
 6743
 6744  021776  100403                        BMI     3$              ;N:C = 0101 ?
 6745  022000  001002                        BNE     3$
 6746  022002  102401                        BVS     3$
 6747  022004  103401                        BCS     4$
 6748
 6749  022006  104001             3$:        ERROR   1               ;COM FAILED TO ALTER THE CODES PROPERLY
 6750  022010  020412             4$:        CMP     R4,(R2)         ;RESULT OK ?
 6751  022012  001402                        BEQ     TST323          ;;BR IF YES
 6752
 6753  022014  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6754  022016  104001             5$:        ERROR   1               ;COM DELIVERED THE WRONG RESULT
 6755
 6756                             ;;*****************************************************************
 6757                             ;*TEST 323      INC DM1 TEST - N:C = 1011
 6758                             ;;*****************************************************************
 6759  022020                     TST323:
 6760  022020  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6761  022022  012700  000323                MOV     #323,R0         ;;LOAD R0 WITH TEST NUMBER
 6762  022026  013701  022050                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6763  022032  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6764  022036  005004                        CLR     R4              ;RESULT S / B = 000000
 6765  022040  012712  177777                MOV     #-1,(R2)                ;[DEST] = 177777
 6766  022044  000257                        CCC                     ;CLEAR CODES
 6767  022046  000273                        273                     ;N:C = 1011
 6768
 6769  022050  005212             2$:        INC     (R2)            ;TEST THE INC
 6770
 6771  022052  100403                        BMI     3$              ;N:C = 0101 ?
 6772  022054  001002                        BNE     3$
 6773  022056  102401                        BVS     3$
 6774  022060  103401                        BCS     4$
 6775
 6776  022062  104001             3$:        ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
 6777  022064  020412             4$:        CMP     R4,(R2)         ;RESULT OK ?
 6778  022066  001402                        BEQ     TST324          ;;BR IF YES
 6779
 6780  022070  011203                        MOV     (R2),R3         ;GET THE WAS DATA
 6781  022072  104001             5$:        ERROR   1               ;INC DELIVERED THE WRONG RESULT
 6782
 6783                             ;;*****************************************************************
 6784                             ;*TEST 324      INC DM1 TEST - N:C = 0100
 6785                             ;;*****************************************************************
 6786  022074                     TST324:
 6787  022074  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6788  022076  012700  000324                MOV     #324,R0         ;;LOAD R0 WITH TEST NUMBER
 6789  022102  013701  022126                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6790  022106  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR   MBUF0
```

```
 6791  022112  012704  100000              MOV     #100000,R4      ;RESULT S / B = 100000
 6792  022116  012712  077777              MOV     #77777,(R2)     ;[DEST] = 77777
 6793  022122  000257                      CCC                     ;CLEAR CODES
 6794  022124  000264                      264                     ;N:C = 0100
 6795
 6796  022126  005212              2$:     INC     (R2)            ;TEST THE INC
 6797
 6798  022130  100003                      BPL     3$              ;N:C = 1010 ?
 6799  022132  001402                      BEQ     3$
 6800  022134  102001                      BVC     3$
 6801  022136  103001                      BCC     4$
 6802
 6803  022140  104001              3$:     ERROR   1               ;INC FAILED TO ALTER THE CODES PROPERLY
 6804  022142  020412              4$:     CMP     R4,(R2)         ;RESULT OK ?
 6805  022144  001402                      BEQ     TST325          ;;BR IF YES
 6806
 6807  022146  011203                      MOV     (R2),R3         ;GET THE WAS DATA
 6808  022150  104001              5$:     ERROR   1               ;INC DELIVERED THE WRONG RESULT
 6809
 6810                              ;;*****************************************************************
 6811                              ;*TEST 325       DEC DM1 TEST - N:C = 1011
 6812                              ;;*****************************************************************
 6813  022152                      TST325:
 6814  022152  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6815  022154  012700  000325              MOV     #325,R0         ;;LOAD R0 WITH TEST NUMBER
 6816  022160  013701  022202              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6817  022164  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6818  022170  005004                      CLR     R4              ;RESULT S / B = 000000
 6819  022172  012712  000001              MOV     #1,(R2)         ;[DEST] = 1
 6820  022176  000257                      CCC                     ;CLEAR CODES
 6821  022200  000273                      273                     ;N:C = 1011
 6822
 6823  022202  005312              2$:     DEC     (R2)            ;TEST THE DEC
 6824
 6825  022204  100403                      BMI     3$              ;N:C = 0101 ?
 6826  022206  001002                      BNE     3$
 6827  022210  102401                      BVS     3$
 6828  022212  103401                      BCS     4$
 6829
 6830  022214  104001              3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6831  022216  020412              4$:     CMP     R4,(R2)         ;RESULT OK ?
 6832  022220  001402                      BEQ     TST326          ;;BR IF YES
 6833
 6834  022222  011203                      MOV     (R2),R3         ;GET THE WAS DATA
 6835  022224  104001              5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6836
 6837                              ;;*****************************************************************
 6838                              ;*TEST 326       DEC DM1 TEST - N:C = 1100
 6839                              ;;*****************************************************************
 6840  022226                      TST326:
 6841  022226  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6842  022230  012700  000326              MOV     #326,R0         ;;LOAD R0 WITH TEST NUMBER
 6843  022234  013701  022260              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6844  022240  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6845  022244  012704  077777              MOV     #77777,R4       ;RESULT S / B = 77777
 6846  022250  012712  100000              MOV     #100000,(R2)    ;[DEST] = 100000
```

```
 6847  022254  000257                  CCC                     ;CLEAR CODES
 6848  022256  000274                  274                     ;N:C = 1100
 6849
 6850  022260  005312          2$:     DEC     (R2)            ;TEST THE DEC
 6851
 6852  022262  100403                  BMI     3$              ;N:C = 0010 ?
 6853  022264  001402                  BEQ     3$
 6854  022266  102001                  BVC     3$
 6855  022270  103001                  BCC     4$
 6856
 6857  022272  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6858  022274  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6859  022276  001402                  BEQ     TST327          ;;BR IF YES
 6860
 6861  022300  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6862  022302  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6863
 6864                          ;;**********************************************************************
 6865                          ;*TEST 327      DEC DM1 TEST - N:C = 0000
 6866                          ;;**********************************************************************
 6867  022304                  TST327:
 6868  022304  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6869  022306  012700  000327          MOV     #327,R0         ;;LOAD R0 WITH TEST NUMBER
 6870  022312  013701  022332          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6871  022316  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6872  022322  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
 6873  022326  005012                  CLR     (R2)            ;[DEST] = 000000
 6874  022330  000257                  CCC                     ;CLEAR CODES
 6875
 6876  022332  005312          2$:     DEC     (R2)            ;TEST THE DEC
 6877
 6878  022334  100003                  BPL     3$              ;N:C = 1000 ?
 6879  022336  001402                  BEQ     3$
 6880  022340  102401                  BVS     3$
 6881  022342  103001                  BCC     4$
 6882
 6883  022344  104001          3$:     ERROR   1               ;DEC FAILED TO ALTER THE CODES PROPERLY
 6884  022346  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6885  022350  001402                  BEQ     TST330          ;;BR IF YES
 6886
 6887  022352  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6888  022354  104001          5$:     ERROR   1               ;DEC DELIVERED THE WRONG RESULT
 6889
 6890                          ;;**********************************************************************
 6891                          ;*TEST 330      ASL DM1 TEST - N:C = 1000
 6892                          ;;**********************************************************************
 6893  022356                  TST330:
 6894  022356  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6895  022360  012700  000330          MOV     #330,R0         ;;LOAD R0 WITH TEST NUMBER
 6896  022364  013701  022406          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6897  022370  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6898  022374  005004                  CLR     R4              ;RESULT S / B = 000000
 6899  022376  012712  100000          MOV     #100000,(R2)    ;[DEST] = 100000
 6900  022402  000257                  CCC                     ;CLEAR CODES
 6901  022404  000270                  SEN                     ;N:C = 1000
 6902
```

```
 6903  022406  006312                  2$:     ASL     (R2)            ;TEST THE ASL
 6904
 6905  022410  100403                          BMI     3$              ;N:C = 0111 ?
 6906  022412  001002                          BNE     3$
 6907  022414  102001                          BVC     3$
 6908  022416  103401                          BCS     4$
 6909
 6910  022420  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6911  022422  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 6912  022424  001402                          BEQ     TST331          ;;BR IF YES
 6913
 6914  022426  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 6915  022430  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6916
 6917                                  ;;*********************************************************************
 6918                                  ;*TEST 331       ASL DM1 TEST - N:C = 0101
 6919                                  ;;*********************************************************************
 6920  022432                          TST331:
 6921  022432  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6922  022434  012700  000331                  MOV     #331,R0         ;;LOAD R0 WITH TEST NUMBER
 6923  022440  013701  022464                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6924  022444  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6925  022450  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
 6926  022454  012712  040000                  MOV     #40000,(R2)     ;[DEST] = 40000
 6927  022460  000257                          CCC                     ;CLEAR CODES
 6928  022462  000265                          265                     ;N:C = 0101
 6929
 6930  022464  006312                  2$:     ASL     (R2)            ;TEST THE ASL
 6931
 6932  022466  100003                          BPL     3$              ;N:C = 1010 ?
 6933  022470  001402                          BEQ     3$
 6934  022472  102001                          BVC     3$
 6935  022474  103001                          BCC     4$
 6936
 6937  022476  104001                  3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6938  022500  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
 6939  022502  001402                          BEQ     TST332          ;;BR IF YES
 6940
 6941  022504  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 6942  022506  104001                  5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6943
 6944                                  ;;*********************************************************************
 6945                                  ;*TEST 332       ASL DM1 TEST - N:C = 0010
 6946                                  ;;*********************************************************************
 6947  022510                          TST332:
 6948  022510  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6949  022512  012700  000332                  MOV     #332,R0         ;;LOAD R0 WITH TEST NUMBER
 6950  022516  013701  022536                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6951  022522  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6952  022526  005004                          CLR     R4              ;RESULT S / B = 000000
 6953  022530  005012                          CLR     (R2)            ;[DEST] = 000000
 6954  022532  000257                          CCC                     ;CLEAR CODES
 6955  022534  000262                          SEV                     ;N:C = 0010
 6956
 6957  022536  006312                  2$:     ASL     (R2)            ;TEST THE ASL
 6958
```

```
 6959  022540  100403                  BMI     3$              ;N:C = 0100 ?
 6960  022542  001002                  BNE     3$
 6961  022544  102401                  BVS     3$
 6962  022546  103001                  BCC     4$
 6963
 6964  022550  104001          3$:     ERROR   1               ;ASL FAILED TO ALTER THE CODES PROPERLY
 6965  022552  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6966  022554  001402                  BEQ     TST333          ;;BR IF YES
 6967
 6968  022556  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6969  022560  104001          5$:     ERROR   1               ;ASL DELIVERED THE WRONG RESULT
 6970
 6971                          ;;**********************************************************
 6972                          ;*TEST 333       ROL DM1 TEST - N:C = 1101
 6973                          ;;**********************************************************
 6974  022562                  TST333:
 6975  022562  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 6976  022564  012700  000333          MOV     #333,R0         ;;LOAD R0 WITH TEST NUMBER
 6977  022570  013701  022614          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 6978  022574  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 6979  022600  012704  052525          MOV     #52525,R4       ;RESULT S / B = 52525
 6980  022604  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
 6981  022610  000257                  CCC                     ;CLEAR CODES
 6982  022612  000275                  275                     ;N:C = 1101
 6983
 6984  022614  006112          2$:     ROL     (R2)            ;TEST THE ROL
 6985
 6986  022616  100403                  BMI     3$              ;N:C = 0011 ?
 6987  022620  001402                  BEQ     3$
 6988  022622  102001                  BVC     3$
 6989  022624  103401                  BCS     4$
 6990
 6991  022626  104001          3$:     EPROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
 6992  022630  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
 6993  022632  001402                  BEQ     TST334          ;;BR IF YES
 6994
 6995  022634  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 6996  022636  104001          5$:     ERROR   1               ;ROL DELIVERED THE WRONG RESULT
 6997
 6998                          ;;**********************************************************
 6999                          ;*TEST 334       ROL DM1 TEST - N:C = 0101
 7000                          ;;**********************************************************
 7001  022640                  TST334:
 7002  022640  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7003  022642  012700  000334          MOV     #334,R0         ;;LOAD R0 WITH TEST NUMBER
 7004  022646  013701  022672          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7005  022652  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 7006  022656  012704  125253          MOV     #125253,R4      ;RESULT S / B = 125253
 7007  022662  012712  052525          MOV     #52525,(R2)     ;[DEST] = 52525
 7008  022666  000257                  CCC                     ;CLEAR CODES
 7009  022670  000265                  265                     ;N:C = 0101
 7010
 7011  022672  006112          2$:     ROL     (R2)            ;TEST THE ROL
 7012
 7013  022674  100003                  BPL     3$              ;N:C - 1010 ?
 7014  022676  001402                  BEQ     3$
```

```
                        BVC     3$
                        BCC     4$

3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
4$:     CMP     R4,(R2)         ;RESULT OK ?
        BEQ     TST335          ;;BR IF YES

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

;;*******************************************************************
;*TEST 335       ROL DM1 TEST - N:C = 0010
;;*******************************************************************
TST335:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #335,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
        CLR     R4              ;RESULT S / B = 000000
        CLR     (R2)            ;[DEST] = 000000
        CCC                     ;CLEAR CODES
        SEV                     ;N:C = 0010

2$:     ROL     (R2)            ;TEST THE ROL

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

3$:     ERROR   1               ;ROL FAILED TO ALTER THE CODES PROPERLY
4$:     CMP     R4,(R2)         ;RESULT OK ?
        BEQ     TST336          ;;BR IF YES

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

;;*******************************************************************
;*TEST 336       ADC DM1 TEST - N:C = 0101
;;*******************************************************************
TST336:
        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
        MOV     #336,R0         ;;LOAD R0 WITH TEST NUMBER
        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
        MOV     #100000,R4      ;RESULT S / B = 100000
        MOV     #77777,(R2)     ;[DEST] = 77777
        CCC                     ;CLEAR CODES
        265                     ;N:C = 0101

2$:     ADC     (R2)            ;TEST THE ADC

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

```
                                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
                                4$:     CMP     R4,(R2)         ;RESULT OK ?
                                        BEQ     TST337          ;;BR IF YES

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

                                ;;**************************************************************
                                ;*TEST 337      ADC DM1 TEST - N:C = 1011
                                ;;**************************************************************
                                TST337:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #337,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        CLR     R4              ;RESULT S  /  B - 000000
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;CLEAR CODES
                                        273                     ;N:C = 1011

                                2$:     ADC     (R2)            ;TEST THE ADC

                                        BMI     3$              ;N:C = 0101 ?
                                        BNE     3$
                                        BVS     3$
                                        BCS     4$

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

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

                                ;;**************************************************************
                                ;*TEST 340 -    ADC DM1 TEST - N:C - 1010
                                ;;**************************************************************
                                TST340:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #340,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #-1,R4          ;RESULT S  /  B = 177777
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;CLEAR CODES
                                        272                     ;N:C = 1010

                                2$:     ADC     (R2)            ;TEST THE ADC

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

                                3$:     ERROR   1               ;ADC FAILED TO ALTER THE CODES PROPERLY
```

```
7127  023170  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7128  023172  001402                          BEQ     TST341          ;;BR IF YES
7129
7130  023174  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7131  023176  104001                  5$:     ERROR   1               ;ADC DELIVERED THE WRONG RESULT
7132
7133                                  ;;**********************************************************************
7134                                  ;*TEST 341       SBC DM1 TEST - N:C = 1011
7135                                  ;;**********************************************************************
7136  023200                          TST341:
7137  023200  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7138  023202  012700  000341                  MOV     #341,R0         ;;LOAD R0 WITH TEST NUMBER
7139  023206  013701  023230                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7140  023212  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7141  023216  005004                          CLR     R4              ;RESULT S / B = 000000
7142  023220  012712  000001                  MOV     #1,(R2)         ;[DEST] = +1
7143  023224  000257                          CCC                     ;CLEAR CODES
7144  023226  000273                          273                     ;N:C = 1011
7145
7146  023230  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7147
7148  023232  100403                          BMI     3$              ;N:C - 0100 ?
7149  023234  001002                          BNE     3$
7150  023236  102401                          BVS     3$
7151  023240  103001                          BCC     4$
7152
7153  023242  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7154  023244  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7155  023246  001402                          BEQ     TST342          ;;BR IF YES
7156
7157  023250  011203                          MOV     (R2),R3         ;GET THE WAS DATA
7158  023252  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
7159
7160                                  ;;**********************************************************************
7161                                  ;*TEST 342       SBC DM1 TEST - N:C = 0101
7162                                  ;;**********************************************************************
7163  023254                          TST342:
7164  023254  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7165  023256  012700  000342                  MOV     #342,R0         ;;LOAD R0 WITH TEST NUMBER
7166  023262  013701  023306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7167  023266  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7168  023272  012704  077777                  MOV     #077777,R4      ;RESULT S / B = 077777
7169  023276  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
7170  023302  000257                          CCC                     ;CLEAR CODES
7171  023304  000265                          265                     ;N:C = 0101
7172
7173  023306  005612                  2$:     SBC     (R2)            ;TEST THE SBC
7174
7175  023310  100403                          BMI     3$              ;N:C = 0010 ?
7176  023312  001402                          BEQ     3$
7177  023314  102001                          BVC     3$
7178  023316  103001                          BCC     4$
7179
7180  023320  104001                  3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
7181  023322  020412                  4$:     CMP     R4,(R2)         ;RESULT OK ?
7182  023324  001402                          BEQ     TST343          ;;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 343       SBC DM1 TEST - N:C = 1110
                                ;;*****************************************************************
023332                          TST343:
023332  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023334  012700  000343          MOV     #343,R0         ;;LOAD R0 WITH TEST NUMBER
023340  013701  023364          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023344  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023350  012704  000001          MOV     #1,R4           ;RESULT S / B = 1
023354  012712  000001          MOV     #1,(R2)         ;[DEST] = 1
023360  000257                  CCC                     ;CLEAR CODES
023362  000276                  276                     ;N:C = 1110
023364  005612          2$:     SBC     (R2)            ;TEST THE SBC
023366  100403                  BMI     3$              ;N:C - 0000 ?
023370  001402                  BEQ     3$
023372  102401                  BVS     3$
023374  103001                  BCC     4$
023376  104001          3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023400  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
023402  001402                  BEQ     TST344          ;;BR IF YES
023404  011203                  MOV     (R2),R3         ;GET THE WAS DATA
023406  104001          5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
                                ;;*****************************************************************
                                ;*TEST 344       SBC DM1 TEST - N:C = 0111
                                ;;*****************************************************************
023410                          TST344:
023410  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
023412  012700  000344          MOV     #344,R0         ;;LOAD R0 WITH TEST NUMBER
023416  013701  023440          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
023422  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
023426  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
023432  005012                  CLR     (R2)            ;[DEST] = 000000
023434  000257                  CCC                     ;CLEAR CODES
023436  000267                  267                     ;N:C = 0111
023440  005612          2$:     SBC     (R2)            ;TEST THE SBC
023442  100003                  BPL     3$              ;N:C = 1001 ?
023444  001402                  BEQ     3$
023446  102401                  BVS     3$
023450  103401                  BCS     4$
023452  104001          3$:     ERROR   1               ;SBC FAILED TO ALTER THE CODES PROPERLY
023454  020412          4$:     CMP     R4,(R2)         ;RESULT OK ?
023456  001402                  BEQ     TST345          ;;BR IF YES
023460  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
 7239  023462  104001                  5$:     ERROR   1               ;SBC DELIVERED THE WRONG RESULT
 7240
 7241                                  ;;*****************************************************************
 7242                                  ;*TEST 345       NEGB - MODE 0 TEST - N:C = 0110
 7243                                  ;;*****************************************************************
 7244  023464                          TST345:
 7245  023464  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7246  023466  012700  000345                 MOV     #345,R0         ;;LOAD R0 WITH TEST NUMBER
 7247  023472  013701  023512                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7248  023476  012704  177776                 MOV     #177776,R4      ;RESULT S / B = 376 (LO BYTE)
 7249  023502  012703  177402                 MOV     #177402,R3      ;[DEST] = 177402
 7250  023506  000257                         CCC                     ;CLEAR FLAGS
 7251  023510  000266                         266                     ;N:C = 0110
 7252
 7253  023512  105403                  2$:     NEGB    R3              ;TEST THE NEGB
 7254
 7255  023514  100003                          BPL     3$              ;N:C - 1001
 7256  023516  001402                          BEQ     3$
 7257  023520  102401                          BVS     3$
 7258  023522  103401                          BCS     4$
 7259
 7260  023524  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
 7261
 7262  023526  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
 7263  023530  001401                          BEQ     TST346          ;;BR IF YES
 7264
 7265  023532  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
 7266
 7267                                  ;;*****************************************************************
 7268                                  ;*TEST 346       NEGB - MODE 0 TEST - N:C = 0011
 7269                                  ;;*****************************************************************
 7270  023534                          TST346:
 7271  023534  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7272  023536  012700  000346                 MOV     #346,R0         ;;LOAD R0 WITH TEST NUMBER
 7273  023542  013701  023562                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7274  023546  012704  177400                 MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
 7275  023552  012703  177400                 MOV     #177400,R3      ;[DEST] = 177400
 7276  023556  000257                         CCC                     ;CLEAR FLAGS
 7277  023560  000263                         263                     ;N:C = 0011
 7278
 7279  023562  105403                  2$:     NEGB    R3              ;TEST THE NEGB
 7280
 7281  023564  100403                          BMI     3$              ;N:C = 0100
 7282  023566  001002                          BNE     3$
 7283  023570  102401                          BVS     3$
 7284  023572  103001                          BCC     4$
 7285
 7286  023574  104002                  3$:     ERROR   2               ;NEGB FAILED TO ALTER CODES PROPERLY
 7287
 7288  023576  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
 7289  023600  001401                          BEQ     TST347          ;;BR IF YES
 7290
 7291  023602  104002                  5$:     ERROR   2               ;NEGB DELIVERED THE WRONG RESULT
 7292
 7293                                  ;;*****************************************************************
 7294                                  ;*TEST 347       NEGB - MODE 0 TEST - N:C = 1101
```

```
                                ;;*****************************************************************
                                TST347:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #347,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177600,R4      ;RESULT S / B = 200 (LO BYTE)
                                        MOV     #177600,R3      ;[DEST] = 177600
                                        CCC                     ;CLEAR FLAGS
                                        275                     ;N:C = 1101
                                
023632  105403                  2$:   NEGB    R3                ;TEST THE NEGB
                                
023634  100003                        BPL     3$              ;N:C = 1011
023636  001402                        BEQ     3$
023640  102001                        BVC     3$
023642  103401                        BCS     4$
                                
023644  104002                  3$:   ERROR   2                 ;NEGB FAILED TO ALTER CODES PROPERLY
                                
023646  020403                  4$:   CMP     R4,R3             ;CORRECT RESULT ?
023650  001401                        BEQ     TST350          ;;BR IF YES
                                
023652  104002                  5$:   ERROR   2                 ;NEGB DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 350      CLRB - MODE 0 TEST - N:C = 1011
                                ;;*****************************************************************
                                TST350:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #350,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #177400,R4      ;RESULT S / B = 000 (LO BYTE)
                                        MOV     #-1,R3          ;[DEST] = 177777
                                        CCC                     ;CLEAR FLAGS
                                        273                     ;N:C = 1011
                                
023702  105003                  2$:   CLRB    R3                ;TEST THE CLRB
                                
023704  100403                        BMI     3$              ;N:C = 0100 ?
023706  001002                        BNE     3$
023710  102401                        BVS     3$
023712  103001                        BCC     4$
                                
023714  104002                  3$:   ERROR   2                 ;CLRB FAILED TO SET CODES PROPERLY
                                
023716  020403                  4$:   CMP     R4,R3             ;RESULT CORRECT ?
023720  001401                        BEQ     TST351          ;;BR IF YES
                                
023722  104002                  5$:   ERROR   2                 ;CLRB DELIVERED THE WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 351      CLRB - MODE 0 TEST - N:C = 0100
                                ;;*****************************************************************
                                TST351:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #351,R0         ;;LOAD R0 WITH TEST NUMBER
```

```
 7351  023732  013701  023752          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7352  023736  012704  177400          MOV     #177400,R4      ;RESULT S / B = 000 (.O BYTE)
 7353  023742  012703  177777          MOV     #-1,R3          ;[DEST] = 177777
 7354  023746  000257                  CCC                     ;CLEAR FLAGS
 7355  023750  000264                  SEZ                     ;N:C = 0100
 7356
 7357  023752  105003          2$:     CLRB    R3              ;TEST THE CLRB
 7358
 7359  023754  100403                  BMI     3$              ;N:C = 0100 ?
 7360  023756  001002                  BNE     3$
 7361  023760  102401                  BVS     3$
 7362  023762  103001                  BCC     4$
 7363
 7364  023764  104002          3$:     ERROR   2               ;CLRB FAILED TO SET CODES PROPERLY
 7365
 7366  023766  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
 7367  023770  001401                  BEQ     TST352          ;;BR IF YES
 7368
 7369  023772  104002          5$:     ERROR   2               ;CLRB DELIVERED THE WRONG RESULT
 7370
 7371                          ;;*****************************************************************
 7372                          ;*TEST 352       CLRB TEST - DM2 - ODD ADDRESS
 7373                          ;;*****************************************************************
 7374  023774                  TST352:
 7375  023774  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7376  023776  012700  000352          MOV     #352,R0         ;;LOAD R0 WITH TEST NUMBER
 7377  024002  013701  024032          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7378  024006  012702  063313          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
 7379  024012  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 7380  024016  012705  063312          MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
 7381  024022  010203                  MOV     R2,R3           ;R3 CONTAINS DEST ADDR
 7382  024024  012715  177777          MOV     #-1,(R5)        ;[DEST] = 177777
 7383  024030  000257                  CCC                     ;SCOPE SYNC
 7384
 7385  024032  105023          2$:     CLRB    (R3)+           ;TEST THE CLRB
 7386
 7387  024034  022703  063314          CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
 7388  024040  001401                  BEQ     4$              ;BR IF YES
 7389
 7390  024042  104005          3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG
 7391
 7392  024044  020415          4$:     CMP     R4,(R5)         ;CORRECT RESULT ?
 7393  024046  001402                  BEQ     TST353          ;;BR IF YES
 7394
 7395  024050  011503                  MOV     (R5),R3         ;GET THE WAS DATA
 7396  024052  104001          5$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT
 7397
 7398                          ;;*****************************************************************
 7399                          ;*TEST 353       CLRB TEST - DM1 - ODD ADDRESS
 7400                          ;;*****************************************************************
 7401  024054                  TST353:
 7402  024054  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 7403  024056  012700  000353          MOV     #353,R0         ;;LOAD R0 WITH TEST NUMBER
 7404  024062  013701  024112          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 7405  024066  012702  063313          MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
 7406  024072  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
```

```
                                        MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
                                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
                                        MOV     #-1,(R5)        ;[DEST] = 177777
                                        CCC                     ;SCOPE SYNC

                                2$:     CLRB    (R3)            ;TEST THE CLRB

                                        CMP     R4,(R5)         ;CORRECT RESULT ?
                                        BEQ     TST354          ;;BR IF YES

                                        MOV     (R5),R3         ;GET THE WAS DATA
                                3$:     ERROR   1               ;CLRB DELIVERED WRONG RESULT

                                ;;*******************************************************************
                                ;*TEST 354       CLRB TEST - DM2 - EVEN ADDRESS
                                ;;*******************************************************************
                                TST354:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #354,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #177400,R4      ;RESULT S / B = 177400
                                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;SCOPE SYNC

                                2$:     CLRB    (R3)+           ;TEST THE CLRB

                                        CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
                                        BEQ     4$              ;BR IF YES

                                3$:     ERROR   5               ;CLRB FAILED TO UPDATE DEST REG

                                4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
                                        BEQ     TST355          ;;BR IF YES

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

                                ;;*******************************************************************
                                ;*TEST 355       CLRB TEST - DM1 - EVEN ADDRESS
                                ;;*******************************************************************
                                TST355:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
                                        MOV     #355,R0         ;;LOAD R0 WITH TEST NUMBER
                                        MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                        MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
                                        MOV     #177400,R4      ;RESULT S / B = 177400
                                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
                                        MOV     #-1,(R2)        ;[DEST] = 177777
                                        CCC                     ;SCOPE SYNC

                                2$:     CLRB    (R3)            ;TEST THE CLRB

                                        CMP     R4,(R2)         ;CORRECT RESULT ?
                                        BEQ     TST356          ;;BR IF YES
```

```
                                        MOV     (R2),R3         ;GET THE WAS DATA
024242  104001                  3$:   ERROR   1               ;CLRB DELIVERED WRONG RESULT

                                ;;**********************************************************
                                ;*TEST 356       NEGB TEST - DM2 - ODD ADDRESS
                                ;;**********************************************************
024244                          TST356:
024244  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
024246  012700  000356                MOV     #356,R0         ;;LOAD P0 WITH TEST NUMBER
024252  013701  024302                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
024256  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
024262  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
024266  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
024272  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
024274  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
024300  000257                        CCC                     ;SCOPE SYNC

024302  105423                  2$:   NEGB    (R3)+           ;TEST THE NEGB

024304  022703  063314                CMP     #MBUF0+2,R3     ;DID DEST REG GET INCREMENTED ?
024310  001401                        BEQ     4$              ;BR IF YES

024312  104005                  3$:   ERROR   5               ;NEGB FAILED TO UPDATE DEST REG

024314  020415                  4$:   CMP     R4,(R5)         ;CORRECT RESULT ?
024316  001402                        BEQ     TST357          ;;BR IF YES

024320  011503                        MOV     (R5),R3         ;GET THE WAS DATA
024322  104001                  5$:   ERROR   1               ;NEGB DELIVERED WRONG RESULT

                                ;;**********************************************************
                                ;*TEST 357       NEGB TEST - DM1 - ODD ADDRESS
                                ;;**********************************************************
024324                          TST357:
024324  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
024326  012700  000357                MOV     #357,R0         ;;LOAD R0 WITH TEST NUMBER
024332  013701  024374                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
                                .SBTTL USER CONTROLLED BREAKPOINT -- BIT7
024336  032737  000200  063234        BIT     #BIT7,@#BPTLOC  ;BREAKPOINT HALT SET ??
024344  001401                        BEQ     .+4             ;BR IF NOT
024346  000000                        HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
024350  012702  063313                MOV     #MBUF0+1,R2     ;DEST ADDR = MBUF0+1
024354  012704  000777                MOV     #777,R4         ;RESULT S / B = 777
024360  012705  063312                MOV     #MBUF0,R5       ;POINT R5 TO CHECK RESULT
024364  010203                        MOV     R2,R3           ;R3 CONTAINS DEST ADDR
024366  012715  177777                MOV     #-1,(R5)        ;[DEST] = 177777
024372  000257                        CCC                     ;SCOPE SYNC

024374  105413                  2$:   NEGB    (R3)            ;TEST THE NEGB

024376  020415                        CMP     R4,(R5)         ;CORRECT RESULT ?
024400  001402                        BEQ     TST360          ;;BR IF YES

024402  011503                        MOV     (R5),R3         ;GET THE WAS DATA
```

```
7518  024404  104001              3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7519
7520                              ;;*****************************************************************
7521                              ;*TEST 360       NEGB TEST - DM2 - EVEN ADDRESS
7522                              ;;*****************************************************************
7523  024406                      TST360:
7524  024406  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7525  024410  012700  000360              MOV     #360,R0         ;;LOAD R0 WITH TEST NUMBER
7526  024414  013701  024440              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7527  024420  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7528  024424  012704  177401              MOV     #177401,R4      ;RESULT S / B = 177401
7529  024430  010203                      MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7530  024432  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
7531  024436  000257                      CCC                     ;SCOPE SYNC
7532
7533  024440  105423              2$:     NEGB    (R3)+           ;TEST THE NEGB
7534
7535  024442  022703  063313              CMP     #MBUF0+1,R3     ;DID DEST REG GET INCREMENTED ?
7536  024446  001401                      BEQ     4$              ;BR IF YES
7537
7538  024450  104005              3$:     ERROR   5               ;NEGB FAILED TO UPDATE DEST REG
7539
7540  024452  020412              4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
7541  024454  001402                      BEQ     TST361          ;;BR IF YES
7542
7543  024456  011203                      MOV     (R2),R3         ;GET THE WAS DATA
7544  024460  104001              5$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7545
7546                              ;;*****************************************************************
7547                              ;*TEST 361       NEGB TEST - DM1 - EVEN ADDRESS
7548                              ;;*****************************************************************
7549  024462                      TST361:
7550  024462  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7551  024464  012700  000361              MOV     #361,R0         ;;LOAD R0 WITH TEST NUMBER
7552  024470  013701  024514              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7553  024474  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7554  024500  012704  177401              MOV     #177401,R4      ;RESULT S / B = 177401
7555  024504  010203                      MOV     R2,R3           ;R3 CONTAINS DEST ADDR
7556  024506  012712  177777              MOV     #-1,(R2)        ;[DEST] = 177777
7557  024512  000257                      CCC                     ;SCOPE SYNC
7558
7559  024514  105413              2$:     NEGB    (R3)            ;TEST THE NEGB
7560
7561  024516  020412                      CMP     R4,(R2)         ;CORRECT RESULT ?
7562  024520  001402                      BEQ     TST362          ;;BR IF YES
7563
7564  024522  011203                      MOV     (R2),R3         ;GET THE WAS DATA
7565  024524  104001              3$:     ERROR   1               ;NEGB DELIVERED WRONG RESULT
7566
7567                              ;;*****************************************************************
7568                              ;*TEST 362       ADD TEST - SM0,DM0 - N:C - 1010
7569                              ;;*****************************************************************
7570  024526                      TST362:
7571  024526  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7572  024530  012700  000362              MOV     #362,R0         ;;LOAD R0 WITH TEST NUMBER
7573  024534  013701  024556              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
7574  024540  005004                  CLR     R4              ;RESULT S / B = 000000
7575  024542  012705  177777          MOV     #-1,R5          ;SRC OPR = 177777
7576  024546  012703  000001          MOV     #+1,R3          ;[DEST] = +1
7577  024552  000257                  CCC                     ;CLEAR FLAGS
7578  024554  000272                  272                     ;N:C = 1010
7579
7580  024556  060503          2$:     ADD     R5,R3           ;TEST THE ADD
7581
7582  024560  100403                  BMI     3$              ;N:C = 0101
7583  024562  001002                  BNE     3$
7584  024564  102401                  BVS     3$
7585  024566  103401                  BCS     4$
7586
7587  024570  104002          3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7588
7589  024572  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
7590  024574  001401                  BEQ     TST363          ;;BR IF YES
7591
7592  024576  104002          5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7593
7594                          ;;*************************************************************
7595                          ;*TEST 363       ADD TEST - SM0,DM0 - N:C = 0101
7596                          ;;*************************************************************
7597  024600                  TST363:
7598  024600  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7599  024602  012700  000363          MOV     #363,R0         ;;LOAD R0 WITH TEST NUMBER
7600  024606  013701  024632          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7601  024612  012704  100006          MOV     #100006,R4      ;RESULT S / B = 100006
7602  024616  012705  077777          MOV     #77777,R5       ;SRC OPR = 77777
7603  024622  012703  000007          MOV     #7,R3           ;[DEST] = 7
7604  024626  000257                  CCC                     ;CLEAR FLAGS
7605  024630  000265                  265                     ;N:C = 0101
7606
7607  024632  060503          2$:     ADD     R5,R3           ;TEST THE ADD
7608
7609  024634  100003                  BPL     3$              ;N:C = 1010
7610  024636  001402                  BEQ     3$
7611  024640  102001                  BVC     3$
7612  024642  103001                  BCC     4$
7613
7614  024644  104002          3$:     ERROR   2               ;ADD FAILED TO ALTER CODES PROPERLY
7615
7616  024646  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
7617  024650  001401                  BEQ     TST364          ;;BR IF YES
7618
7619  024652  104002          5$:     ERROR   2               ;ADD DELIVERED THE WRONG RESULT
7620
7621                          ;;*************************************************************
7622                          ;*TEST 364       ADD SM1,DM0 TEST
7623                          ;;*************************************************************
7624  024654                  TST364:
7625  024654  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7626  024656  012700  000364          MOV     #364,R0         ;;LOAD R0 WITH TEST NUMBER
7627  024662  013701  024702          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7628  024666  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7629  024672  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
```

```
7630  024676  005003                  CLR     R3              ;[DEST] = 0
7631  024700  000257                  CCC                     ;SCOPE SYNC
7632
7633  024702  061503          2$:     ADD     (R5),R3         ;TEST THE ADD - SM1,DMO
7634
7635  024704  020403                  CMP     R4,R3           ;RESULT = #DWTA?
7636  024706  001401                  BEQ     4$              ;BR IF YES
7637
7638  024710  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7639
7640  024712  022705  063276  4$:     CMP     #ATA,R5         ;DID ADD CHANGE REG.
7641  024716  001401                  BEQ     TST365          ;;BR IF NOT
7642
7643  024720  104005          5$:     ERROR   5               ;REG GOT MODIFIED
7644
7645                          ;;*****************************************************************
7646                          ;*TEST 365       ADD SM2,DMO TEST
7647                          ;;*****************************************************************
7648  024722                  TST365:
7649  024722  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7650  024724  012700  000365          MOV     #365,R0         ;;LOAD R0 WITH TEST NUMBER
7651  024730  013701  024750          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7652  024734  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7653  024740  012705  063276          MOV     #ATA,R5         ;SOURCE ADDR = ATA
7654  024744  005003                  CLR     R3              ;[DEST] = 0
7655  024746  000257                  CCC                     ;SCOPE SYNC
7656
7657  024750  062503          2$:     ADD     (R5)+,R3        ;TEST THE ADD - SM2,DMO
7658
7659  024752  020403                  CMP     R4,R3           ;RESULT = #DWTA
7660  024754  001401                  BEQ     4$              ;BR IF YES
7661
7662  024756  104002          3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7663
7664  024760  022705  063300  4$:     CMP     #ATA+2,R5       ;DID ADD AUTO INCREMENT SOURCE REG?
7665  024764  001401                  BEQ     TST366          ;;BR IF YES
7666
7667  024766  104005          5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7668
7669                          ;;*****************************************************************
7670                          ;*TEST 366       ADD SM3,DMO TEST
7671                          ;;*****************************************************************
7672  024770                  TST366:
7673  024770  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7674  024772  012700  000366          MOV     #366,R0         ;;LOAD R0 WITH TEST NUMBER
7675  024776  013701  025022          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7676  025002  012704  063322          MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7677  025006  012705  063306          MOV     #ATA+10,R5      ;R5 POINTS TO SOURCE ADDR
7678  025012  010437  063312          MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7679  025016  005003                  CLR     R3              ;[DEST] = 0
7680  025020  000257                  CCC                     ;SCOPE SYNC
7681
7682  025022  063503          2$:     ADD     @(R5)+,R3       ;TEST THE ADD - SM3,DMO
7683
7684  025024  020437  063312          CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7685  025030  001401                  BEQ     4$              ;BR IF YES
```

```
7686
7687  025032  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7688
7689  025034  022705  063310        4$:     CMP     #ATA+12,R5      ;DID ADD AUTO INCREMENT SOURCE REG?
7690  025040  001401                        BEQ     TST367          ;;BR IF YES
7691
7692  025042  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
7693
7694                                ;;*************************************************************
7695                                ;*TEST 367       ADD SM4,DM0 TEST
7696                                ;;*************************************************************
7697  025044                        TST367:
7698  025044  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7699  025046  012700  000367               MOV     #367,R0         ;;LOAD R0 WITH TEST NUMBER
7700  025052  013701  025072               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7701  025056  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7702  025062  012705  063300               MOV     #ATA+2,R5       ;SOURCE ADDR = ATA
7703  025066  005003                       CLR     R3              ;[DEST] = 0
7704  025070  000257                       CCC                     ;SCOPE SYNC
7705
7706  025072  064503                2$:     ADD     -(R5),R3        ;TEST THE ADD - SM4,DM0
7707
7708  025074  020403                       CMP     R4,R3           ;RESULT = #DWTA?
7709  025076  001401                       BEQ     4$              ;BR IF YES
7710
7711  025100  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7712
7713  025102  022705  063276        4$:     CMP     #ATA,R5         ;DID SOURCE REG GET DECREMENTED?
7714  025106  001401                       BEQ     TST370          ;;BR IF YES
7715
7716  025110  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG
7717
7718                                ;;*************************************************************
7719                                ;*TEST 370       ADD SM5,DM0 TEST
7720                                ;;*************************************************************
7721  025112                        TST370:
7722  025112  000004                       SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7723  025114  012700  000370               MOV     #370,R0         ;;LOAD R0 WITH TEST NUMBER
7724  025120  013701  025144               MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7725  025124  012704  063322               MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7726  025130  012705  063310               MOV     #ATA+12,R5      ;R5 POINTS TO SOURCE ADDR
7727  025134  010437  063312               MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
7728  025140  005003                       CLR     R3              ;[DEST] = 0
7729  025142  000257                       CCC                     ;SCOPE SYNC
7730
7731  025144  065503                2$:     ADD     @-(R5),R3       ;TEST THE ADD - SM5,DM0
7732
7733  025146  020437  063312               CMP     R4,@#MBUF0      ;RESULT = #DWTA?
7734  025152  001401                       BEQ     4$              ;BR IF YES
7735
7736  025154  104002                3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT
7737
7738  025156  022705  063306        4$:     CMP     #ATA+10,R5      ;DID ADD DECREMENT SOURCE REG?
7739  025162  001401                       BEQ     TST371          ;;BR IF YES
7740
7741  025164  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE SOURCE REG.
```

```
                                ;;*****************************************************************
                                ;*TEST 371       ADD SM6,DM0 TEST
                                ;;*****************************************************************
025166                          TST371:
025166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025170  012700  000371                  MOV     #371,R0         ;;LOAD R0 WITH TEST NUMBER
025174  013701  025214                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025200  012704  063312                  MOV     #MBUF0,R4       ;RESULT S / B = MBUF0
025204  012705  063276                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
025210  005003                          CLR     R3              ;[DEST] = 0
025212  000257                          CCC                     ;SCOPE SYNC

025214  066503  000010          2$:     ADD     10(R5),R3       ;TEST THE ADD - SM6,DM0

025220  020403                          CMP     R4,R3           ;RESULT =MBUF0?
025222  001401                          BEQ     TST372          ;;BR IF YES

025224  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 372       ADD SM7,DM0 TEST
                                ;;*****************************************************************
025226                          TST372:
025226  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025230  012700  000372                  MOV     #372,R0         ;;LOAD R0 WITH TEST NUMBER
025234  013701  025260                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025240  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025244  012705  063276                  MOV     #ATA,R5         ;BASE SOURCE ADDR = ATA
025250  010437  063312                  MOV     R4,@#MBUF0      ;[SOURCE] = #DWTA
025254  005003                          CLR     R3              ;[DEST] = 0
025256  000257                          CCC                     ;SCOPE SYNC

025260  067503  000010          2$:     ADD     @10(R5),R3      ;TEST THE ADD - SM7,DM0

025264  020403                          CMP     R4,R3           ;RESULT = #DWTA?
025266  001401                          BEQ     TST373          ;;BR IF YES

025270  104002                  3$:     ERROR   2               ;ADD DELIVERED WRONG RESULT

                                ;;*****************************************************************
                                ;*TEST 373       ADD SM1,DM1 TEST
                                ;;*****************************************************************
025272                          TST373:
025272  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025274  012700  000373                  MOV     #373,R0         ;;LOAD R0 WITH TEST NUMBER
025300  013701  025324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025304  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025310  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025314  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025320  005012                          CLR     (R2)            ;[DEST] = 0
025322  000257                          CCC                     ;SCOPE SYNC

025324  061512                  2$:     ADD     (R5),(R2)       ;TEST THE ADD - SM1,DM1

025326  020412                          CMP     R4,(R2)         ;RESULT   #DWTA?
```

```
7798  025330  001402                    BEQ     TST374          ;;BR IF YES
7799
7800  025332  011203                    MOV     (R2),R3         ;GET WAS DATA
7801  025334  104001              3$:   ERROR   1               ;ADD DELIVERED WRONG RESULT
7802
7803                                    ;;*******************************************************************
7804                                    ;*TEST 374       ADD SM2,DM1 TEST
7805                                    ;;*******************************************************************
7806  025336                            TST374:
7807  025336  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7808  025340  012700  000374            MOV     #374,R0         ;;LOAD R0 WITH TEST NUMBER
7809  025344  013701  025370            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7810  025350  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7811  025354  012704  063322            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7812  025360  012705  063276            MOV     #ATA,R5         ;SOURCE ADDR = ATA
7813  025364  005012                    CLR     (R2)            ;[DEST] = 0
7814  025366  000257                    CCC                     ;SCOPE SYNC
7815
7816  025370  062512              2$:   ADD     (R5)+,(R2)      ;TEST THE ADD - SM2,DM1
7817
7818  025372  020412                    CMP     R4,(R2)         ;RESULT = #DWTA?
7819  025374  001402                    BEQ     TST375          ;;BR IF YES
7820
7821  025376  011203                    MOV     (R2),R3         ;GET WAS DATA
7822  025400  104001              3$:   ERROR   1               ;ADD DELIVERED WRONG RESULT
7823
7824                                    ;;*******************************************************************
7825                                    ;*TEST 375       ADD SM1,DM2 TEST
7826                                    ;;*******************************************************************
7827  025402                            TST375:
7828  025402  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
7829  025404  012700  000375            MOV     #375,R0         ;;LOAD R0 WITH TEST NUMBER
7830  025410  013701  025436            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
7831  025414  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
7832  025420  012704  063322            MOV     #DWTA,R4        ;RESULT S / B = #DWTA
7833  025424  012705  063276            MOV     #ATA,R5         ;SOURCE ADDR = ATA
7834  025430  010203                    MOV     R2,R3           ;[R3] = DEST ADDR
7835  025432  005012                    CLR     (R2)            ;[DEST] = 0
7836  025434  000257                    CCC                     ;SCOPE SYNC
7837
7838  025436  061523              2$:   ADD     (R5),(R3)+      ;TEST THE ADD - SM1,DM2
7839
7840  025440  020412                    CMP     R4,(R2)         ;RESULT = #DWTA?
7841  025442  001406                    BEQ     4$              ;BR IF YES
7842
7843  025444  010337  063316            MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
7844  025450  011203                    MOV     (R2),R3         ;GET WAS DATA
7845  025452  104001              3$:   ERROR   1               ;ADD DELIVERED WRONG RESULT
7846
7847  025454  013703  063316            MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
7848  025460  022703  063314      4$:   CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG
7849  025464  001401                    BEQ     TST376          ;;BR IF YES
7850
7851  025466  104005              5$:   ERROR   5               ;ADD FAILED TO UPDATE DEST REG
7852
7853                                    ;;*******************************************************************
```

```
                                ;*TEST 376      ADD SM2,DM2 TEST
                                ;;**************************************************************
025470                          TST376:
000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025472  012700  000376                  MOV     #376,R0         ;;LOAD R0 WITH TEST NUMBER
025476  013701  025524                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025502  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025506  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025512  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025516  010203                          MOV     R2,R3           ;[R3] = DEST ADDR
025520  005012                          CLR     (R2)            ;[DEST] = 0
025522  000257                          CCC                     ;SCOPE SYNC

025524  062523                  2$:     ADD     (R5)+,(R3)+     ;TEST THE ADD - SM2,DM2

025526  020412                          CMP     R4,(R2)         ;RESULT = #DWTA
025530  001406                          BEQ     4$              ;BR IF YES

025532  010337  063316                  MOV     R3,@#MBUF1      ;SAVE UPDATED DEST ADDR
025536  011203                          MOV     (R2),R3         ;GET WAS DATA
025540  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

025542  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE UPDATED DEST ADDR
025546  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID ADD INCREMENT DEST REG?
025552  001401                          BEQ     TST377          ;;BR IF YES

025554  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG

                                ;;**************************************************************
                                ;*TEST 377      ADD SM1,DM3 TEST
                                ;;**************************************************************
025556                          TST377:
000004                                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
025560  012700  000377                  MOV     #377,R0         ;;LOAD R0 WITH TEST NUMBER
025564  013701  025614                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
025570  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
025574  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
025600  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
025604  012703  063306                  MOV     #ATA+10,R3      ;[R3] = ADDR OF DEST ADDR
025610  005012                          CLR     (R2)            ;[DEST] = 0
025612  000257                          CCC                     ;SCOPE SYNC

025614  061533                  2$:     ADD     (R5),@(R3)+     ;TEST THE ADD - SM1,DM3

025616  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
025620  001406                          BEQ     4$              ;BR IF YES

025622  010337  063316                  MOV     R3,@#MBUF1      ;SAVE R3
025626  011203                          MOV     (R2),R3         ;GET WAS DATA
025630  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT

025632  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE R3
025636  022703  063310          4$:     CMP     #ATA+12,R3      ;DID ADD INCREMENT DEST REG
025642  001401                          BEQ     TST400          ;;BR IF YES

025644  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
```

```
                        ;;*****************************************************************
                        ;*TEST 400       ADD SM2,DM3 TEST
                        ;;*****************************************************************
025646                  TST400:
025646  000004                 SCOPE                          ;CALL THE SCOPE LOOP UTILITY
025650  012700  000400         MOV     #400,R0                ;;LOAD R0 WITH TEST NUMBER
025654  013701  025704         MOV     @#2$,R1                ;LOAD R1 WITH TEST INSTRUCTION WORD
025660  012702  063312         MOV     #MBUF0,R2              ;DEST ADDR = MBUF0
025664  012704  063322         MOV     #DWTA,R4               ;RESULT S / B = #DWTA
025670  012705  063276         MOV     #ATA,R5                ;SOURCE ADDR = ATA
025674  012703  063306         MOV     #ATA+10,R3             ;[R3] = ADDR OF DEST ADDR
025700  005012                 CLR     (R2)                   ;[DEST] = 0
025702  000257                 CCC                            ;SCOPE SYNC

025704  062533         2$:     ADD     (R5)+,@(R3)+           ;TEST THE ADD - SM2,DM3

025706  020412                 CMP     R4,(R2)                ;RESULT = #DWTA?
025710  001406                 BEQ     4$                     ;BR IF YES

025712  010337  063316         MOV     R3,@#MBUF1             ;SAVE R3
025716  011203                 MOV     (R2),R3                ;GET WAS DATA
025720  104001         3$:     ERROR   1                      ;ADD DELIVERED WRONG RESULT

025722  013703  063316         MOV     @#MBUF1,R3             ;RESTORE R3
025726  022703  063310         CMP     #ATA+12,R3             ;DID ADD INCREMENT DEST REG
025732  001401                 BEQ     TST401                 ;;BR IF YES

025734  104005         5$:     ERROR   5                      ;ADD FAILED TO UPDATE DEST REG

                        ;;*****************************************************************
                        ;*TEST 401       ADD SM1,DM4 TEST
                        ;;*****************************************************************
025736                  TST401:
025736  000004                 SCOPE                          ;CALL THE SCOPE LOOP UTILITY
025740  012700  000401         MOV     #401,R0                ;;LOAD R0 WITH TEST NUMBER
025744  013701  025774         MOV     @#2$,R1                ;LOAD R1 WITH TEST INSTRUCTION WORD
025750  012702  063312         MOV     #MBUF0,R2              ;DEST ADDR = MBUF0
025754  012704  063322         MOV     #DWTA,R4               ;RESULT S / B - #DWTA
025760  012705  063276         MOV     #ATA,R5                ;SOURCE ADDR = ATA
025764  012703  063314         MOV     #MBUF0+2,R3            ;R3 POINTS TO DEST ADDR +2
025770  005012                 CLR     (R2)                   ;[DEST] = 0
025772  000257                 CCC                            ;SCOPE SYNC

025774  061543         2$:     ADD     (R5),-(R3)             ;TEST THE ADD - SM1,DM4

025776  020412                 CMP     R4,(R2)                ;RESULT = #DWTA?
026000  001406                 BEQ     4$                     ;BR IF YES

026002  010337  063316         MOV     R3,@#MBUF1             ;SAVE R3
026006  011203                 MOV     (R2),R3                ;GET WAS DATA
026010  104001         3$:     ERROR   1                      ;ADD DELIVERED WRONG RESULT

026012  013703  063316         MOV     @#MBUF1,R3             ;RESTORE R3
026016  020302         4$:     CMP     R3,R2                  ;DID ADD INCREMENT DEST REG
026020  001401                 BEQ     TST402                 ;;BR IF YES
```

```
                                ;;*****************************************************************
                                ;*TEST 402       ADD SM2,DM4 TEST
                                ;;*****************************************************************
026024                          TST402:
026024  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026026  012700  000402                  MOV     #402,R0         ;;LOAD R0 WITH TEST NUMBER
026032  013701  026062                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026036  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026042  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026046  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
026052  012703  063314                  MOV     #MBUF0+2,R3     ;R3 POINTS TO DEST ADDR +2
026056  005012                          CLR     (R2)            ;[DEST] = 0
026060  000257                          CCC                     ;SCOPE SYNC
                                
026062  061543                  2$:     ADD     (R5),-(R3)      ;TEST THE ADD - SM2,DM4
                                
026064  020412                          CMP     R4,(R2)         ;RESULT = #DWTA?
026066  001406                          BEQ     4$              ;BR IF YES
                                
026070  010337  063316                  MOV     R3,@#MBUF1      ;SAVE R3
026074  011203                          MOV     (R2),R3         ;GET WAS DATA
026076  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
026100  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE R3
026104  020302                  4$:     CMP     R3,R2           ;DID ADD INCREMENT DEST REG?
026106  001401                          BEQ     TST403          ;;BR IF YES
                                
026110  104005                  5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
                                
                                ;;*****************************************************************
                                ;*TEST 403       ADD SM1,DM5 TEST
                                ;;*****************************************************************
026112                          TST403:
026112  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
026114  012700  000403                  MOV     #403,R0         ;;LOAD R0 WITH TEST NUMBER
026120  013701  026150                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
026124  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
026130  012704  063322                  MOV     #DWTA,R4        ;RESULT S / B = #DWTA
026134  012705  063276                  MOV     #ATA,R5         ;SOURCE ADDR = ATA
026140  012703  063310                  MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
026144  005012                          CLR     (R2)            ;[DEST] = 0
026146  000257                          CCC                     ;SCOPE SYNC
                                
026150  061553                  2$:     ADD     (R5),@-(R3)     ;TEST THE ADD - SM1,DM5
                                
026152  020412                          CMP     R4,(R2)         ;RESULT - #DWTA?
026154  001406                          BEQ     4$              ;BR IF YES
                                
026156  010337  063316                  MOV     R3,@#MBUF1      ;SAVE R3
026162  011203                          MOV     (R2),R3         ;GET WAS DATA
026164  104001                  3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
                                
026166  013703  063316                  MOV     @#MBUF1,R3      ;RESTORE R3
```

```
8022  026172  022703  063306        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8023  026176  001401                        BEQ     TST404          ;;BR IF YES
8024
8025  026200  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG.
8026
8027                                  ;;*****************************************************************
8028                                  ;*TEST 404       ADD SM2,DM5 TEST
8029                                  ;;*****************************************************************
8030  026202                          TST404:
8031  026202  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8032  026204  012700  000404                MOV     #404,R0         ;;LOAD R0 WITH TEST NUMBER
8033  026210  013701  026240                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8034  026214  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8035  026220  012704  063322                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8036  026224  012705  063276                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8037  026230  012703  063310                MOV     #ATA+12,R3      ;R3 CONTAINS ADDR OF DEST ADDR PLUS 2
8038  026234  005012                        CLR     (R2)            ;[DEST] = 0
8039  026236  000257                        CCC                     ;SCOPE SYNC
8040
8041  026240  062553                2$:     ADD     (R5)+,@-(R3)    ;TEST THE ADD - SM2,DM5
8042
8043  026242  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8044  026244  001406                        BEQ     4$              ;BR IF YES
8045
8046  026246  010337  063316                MOV     R3,@#MBUF1      ;SAVE R3
8047  026252  011203                        MOV     (R2),R3         ;GET WAS DATA
8048  026254  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8049
8050  026256  013703  063316                MOV     @#MBUF1,R3      ;RESTORE R3
8051  026262  022703  063306        4$:     CMP     #ATA+10,R3      ;DID ADD DECREMENT DEST REG?
8052  026266  001401                        BEQ     TST405          ;;BR IF YES
8053
8054  026270  104005                5$:     ERROR   5               ;ADD FAILED TO UPDATE DEST REG
8055
8056                                  ;;*****************************************************************
8057                                  ;*TEST 405       ADD SM1,DM6 TEST
8058                                  ;;*****************************************************************
8059  026272                          TST405:
8060  026272  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8061  026274  012700  000405                MOV     #405,R0         ;;LOAD R0 WITH TEST NUMBER
8062  026300  013701  026330                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8063  026304  012702  063316                MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
8064  026310  012704  063322                MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8065  026314  012705  063276                MOV     #ATA,R5         ;SOURCE ADDR = ATA
8066  026320  012703  063312                MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
8067  026324  005012                        CLR     (R2)            ;[DEST] = 0
8068  026326  000257                        CCC                     ;SCOPE SYNC
8069
8070  026330  061563  000004        2$:     ADD     (R5),4(R3)      ;TEST THE ADD - SM1,DM6
8071
8072  026334  020412                        CMP     R4,(R2)         ;RESULT = #DWTA?
8073  026336  001402                        BEQ     TST406          ;;BR IF YES
8074
8075  026340  011203                        MOV     (R2),R3         ;GET WAS DATA
8076  026342  104001                3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8077
```

```
8078                                    ;;*****************************************************************
8079                                    ;*TEST 406      ADD SM2,DM6 TEST
8080                                    ;;*****************************************************************
8081  026344                            TST406:
8082  026344  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8083  026346  012700  000406                   MOV     #406,R0         ;;LOAD R0 WITH TEST NUMBER
8084  026352  013701  026402                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8085  026356  012702  063316                   MOV     #MBUF0+4,R2     ;DEST ADDR = MBUF0+4
8086  026362  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8087  026366  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
8088  026372  012703  063312                   MOV     #MBUF0,R3       ;[R3] = BASE DEST ADDR
8089  026376  005012                           CLR     (R2)            ;[DEST] = 0
8090  026400  000257                           CCC                     ;SCOPE SYNC
8091
8092  026402  062563  000004           2$:     ADD     (R5)+,4(R3)     ;TEST THE ADD - SM2,DM6
8093
8094  026406  020412                           CMP     R4,(R2)         ;RESULT = #DWTA?
8095  026410  001402                           BEQ     TST407          ;;BR IF YES
8096
8097  026412  011203                           MOV     (R2),R3         ;GET WAS DATA
8098  026414  104001                   3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8099
8100                                    ;;*****************************************************************
8101                                    ;*TEST 407      ADD SM1,DM7 TEST
8102                                    ;;*****************************************************************
8103  026416                            TST407:
8104  026416  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8105  026420  012700  000407                   MOV     #407,R0         ;;LOAD R0 WITH TEST NUMBER
8106  026424  013701  026452                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8107  026430  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8108  026434  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8109  026440  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
8110  026444  010503                           MOV     R5,R3           ;BASE DEST ADDR = ATA
8111  026446  005012                           CLR     (R2)            ;[DEST] = 0
8112  026450  000257                           CCC                     ;SCOPE SYNC
8113
8114  026452  061573  000010           2$:     ADD     (R5),@10(R3)    ;TEST THE ADD - SM1,DM7
8115
8116  026456  020412                           CMP     R4,(R2)         ;RESULT = #DWTA?
8117  026460  001402                           BEQ     TST410          ;;BR IF YES
8118
8119  026462  011203                           MOV     (R2),R3         ;GET WAS DATA
8120  026464  104001                   3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8121
8122                                    ;;*****************************************************************
8123                                    ;*TEST 410      ADD SM2,DM7 TEST
8124                                    ;;*****************************************************************
8125  026466                            TST410:
8126  026466  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8127  026470  012700  000410                   MOV     #410,R0         ;;LOAD R0 WITH TEST NUMBER
8128  026474  013701  026522                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8129  026500  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8130  026504  012704  063322                   MOV     #DWTA,R4        ;RESULT S / B = #DWTA
8131  026510  012705  063276                   MOV     #ATA,R5         ;SOURCE ADDR = ATA
8132  026514  010503                           MOV     R5,R3           ;BASE DEST ADDR - ATA
8133  026516  005012                           CLR     (R2)            ;[DEST] = 0
```

```
8134  026520  000257                  CCC                     ;SCOPE SYNC
8135
8136  026522  062573  000010  2$:     ADD     (R5)+,@10(R3)   ;TEST THE ADD - SM2,DM7
8137
8138  026526  020412                  CMP     R4,(R2)         ;RESULT = #DWTA?
8139  026530  001402                  BEQ     TST411          ;;BR IF YES
8140
8141  026532  011203                  MOV     (R2),R3         ;GET WAS DATA
8142  026534  104001          3$:     ERROR   1               ;ADD DELIVERED WRONG RESULT
8143
8144                          ;;**************************************************************
8145                          ;*TEST 411      "XOR RA,RB" TEST - A=B=000000 N:C='010
8146                          ;;**************************************************************
8147  026536                  TST411:
8148  026536  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8149  026540  012700  000411          MOV     #411,R0         ;;LOAD R0 WITH TEST NUMBER
8150  026544  013701  026560          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8151  026550  005004                  CLR     R4              ;RESULT AND MASK = 000000
8152  026552  005003                  CLR     R3              ;[DEST] = 000000
8153  026554  000257                  CCC                     ;SCOPE SYNC
8154  026556  000272                  272                     ;MAKE N:C='010
8155
8156  026560  074403          2$:     XOR     R4,R3           ;TEST THE XOR
8157
8158  026562  100403                  BMI     3$              ;N:C=0100 ??
8159  026564  001002                  BNE     3$
8160  026566  102401                  BVS     3$
8161  026570  103001                  BCC     4$
8162
8163  026572  104002          3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
8164
8165  026574  020403          4$:     CMP     R4,R3           ;RESULT CORRECT?
8166  026576  001401                  BEQ     TST412          ;;BR IF YES
8167
8168  026600  104002          5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
8169
8170                          ;;**************************************************************
8171                          ;*TEST 412      "XOR RA,RB" TEST - A=B=177777 N:C=0101
8172                          ;;**************************************************************
8173  026602                  TST412:
8174  026602  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8175  026604  012700  000412          MOV     #412,R0         ;;LOAD R0 WITH TEST NUMBER
8176  026610  013701  026630          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8177  026614  005004                  CLR     R4              ;RESULT  = 000000
8178  026616  012705  177777          MOV     #-1,R5          ;MASK = 177777
8179  026622  010503                  MOV     R5,R3           ;[DEST]=177777
8180  026624  000257                  CCC                     ;SCOPE SYNC
8181  026626  000265                  265                     ;MAKE N:C=0101
8182
8183  026630  074503          2$:     XOR     R5,R3           ;TEST THE XOR
8184
8185  026632  100403                  BMI     3$              ;N:C=0101 ??
8186  026634  001002                  BNE     3$
8187  026636  102401                  BVS     3$
8188  026640  103401                  BCS     4$
8189
```

```
 8190  026642  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
 8191
 8192  026644  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
 8193  026646  001401                          BEQ     TST413          ;;BR IF YES
 8194
 8195  026650  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
 8196
 8197                                  ;;*********************************************************
 8198                                  ;*TEST 413       "XOR RA,RB" TEST - A=125252,B=052525 N:C-0110
 8199                                  ;;*********************************************************
 8200  026652                          TST413:
 8201  026652  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8202  026654  012700  000413                  MOV     #413,R0         ;;LOAD R0 WITH TEST NUMBER
 8203  026660  013701  026704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8204  026664  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
 8205  026670  012705  125252                  MOV     #125252,R5      ;MASK=125252
 8206  026674  012703  052525                  MOV     #052525,R3      ;[DEST] = 052525
 8207  026700  000257                          CCC                     ;SCOPE SYNC
 8208  026702  000266                          266                     ;MAKE N:C-0110
 8209
 8210  026704  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
 8211
 8212  026706  100003                          BPL     3$              ;N:C=1000 ??
 8213  026710  001402                          BEQ     3$
 8214  026712  102401                          BVS     3$
 8215  026714  103001                          BCC     4$
 8216
 8217  026716  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
 8218
 8219  026720  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT?
 8220  026722  001401                          BEQ     TST414          ;;BR IF YES
 8221
 8222  026724  104002                  5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
 8223
 8224                                  ;;*********************************************************
 8225                                  ;*TEST 414       "XOR RA,RB" TEST - A-052525,B-125252 N:C=1001
 8226                                  ;;*********************************************************
 8227  026726                          TST414:
 8228  026726  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8229  026730  012700  000414                  MOV     #414,R0         ;;LOAD R0 WITH TEST NUMBER
 8230  026734  013701  026760                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8231  026740  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
 8232  026744  012705  052525                  MOV     #52525,R5       ;MASK=052525
 8233  026750  012703  125252                  MOV     #125252,R3      ;[DEST] = 125252
 8234  026754  000257                          CCC                     ;SCOPE SYNC
 8235  026756  000271                          271                     ;MAKE N:C=1001
 8236
 8237  026760  074503                  2$:     XOR     R5,R3           ;TEST THE XOR
 8238
 8239  026762  100003                          BPL     3$              ;N:C=1001 ??
 8240  026764  001402                          BEQ     3$
 8241  026766  102401                          BVS     3$
 8242  026770  103401                          BCS     4$
 8243
 8244  026772  104002                  3$:     ERROR   2               ;XOR FAILED TO SET FLAGS PROPERLY
 8245
```

```
 8246  026774  020403                 4$:     CMP     R4,R3           ;RESULT CORRECT?
 8247  026776  001401                         BEQ     TST415          ;;BR IF YES
 8248
 8249  027000  104002                 5$:     ERROR   2               ;XOR DELIVERED THE WRONG RESULT
 8250
 8251                                 ;;**************************************************************
 8252                                 ;*TEST 415       "XOR RA,(RB)" TEST - A=B=000000 N:C=1010
 8253                                 ;;**************************************************************
 8254  027002                         TST415:
 8255  027002  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8256  027004  012700  000415                 MOV     #415,R0         ;;LOAD R0 WITH TEST NUMBER
 8257  027010  013701  027032                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8258  027014  005004                         CLR     R4              ;RESULT S / B = 000000
 8259  027016  005005                         CLR     R5              ;MASK = 000000
 8260  027020  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8261  027024  005012                         CLR     (R2)            ;[DEST] = 000000
 8262  027026  000257                         CCC                     ;SCOPE SYNC
 8263  027030  000272                         272                     ;MAKE N:C=1010
 8264
 8265  027032  074512                 2$:     XOR     R5,(R2)         ;TEST THE XOR
 8266
 8267  027034  100403                         BMI     3$              ;N:C = 0100 ??
 8268  027036  001002                         BNE     3$
 8269  027040  102401                         BVS     3$
 8270  027042  103001                         BCC     4$
 8271
 8272  027044  104001                 3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
 8273
 8274  027046  020412                 4$:     CMP     R4,(R2)         ;RESULT CORRECT?
 8275  027050  001402                         BEQ     TST416          ;;BR IF YES
 8276
 8277  027052  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 8278  027054  104001                 5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
 8279
 8280                                 ;;**************************************************************
 8281                                 ;*TEST 416       "XOR RA,(RB)" TEST - A=B=177777 N:C=0101
 8282                                 ;;**************************************************************
 8283  027056                         TST416:
 8284  027056  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8285  027060  012700  000416                 MOV     #416,R0         ;;LOAD R0 WITH TEST NUMBER
 8286  027064  013701  027112                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8287  027070  005004                         CLR     R4              ;RESULT S / B = 000000
 8288  027072  012705  177777                 MOV     #-1,R5          ;MASK = 177777
 8289  027076  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8290  027102  012712  177777                 MOV     #-1,(R2)        ;[DEST] = 177777
 8291  027106  000257                         CCC                     ;SCOPE SYNC
 8292  027110  000265                         265                     ;MAKE N:C=0101
 8293
 8294  027112  074512                 2$:     XOR     R5,(R2)         ;TEST THE XOR
 8295
 8296  027114  100403                         BMI     3$              ;N:C = 0101 ??
 8297  027116  001002                         BNE     3$
 8298  027120  102401                         BVS     3$
 8299  027122  103401                         BCS     4$
 8300
 8301  027124  104001                 3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
```

```
8302
8303  027126  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8304  027130  001402                          BEQ     TST417          ;;BR IF YES
8305
8306  027132  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8307  027134  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8308
8309                                  ;;**************************************************************
8310                                  ;*TEST 417      "XOR RA,(RB)" TEST - A=125252,B=052525 N:C=0110
8311                                  ;;**************************************************************
8312  027136                          TST417:
8313  027136  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8314  027140  012700  000417                  MOV     #417,R0         ;;LOAD R0 WITH TEST NUMBER
8315  027144  013701  027206                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8316                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT8
8317  027150  032737  000400  063234          BIT     #BIT8,@#BPTLOC  ;BREAKPOINT HALT SET ??
8318  027156  001401                          BEQ     .+4             ;BR IF NOT
8319  027160  000000                          HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
8320  027162  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8321  027166  012705  125252                  MOV     #125252,R5      ;MASK = 125252
8322  027172  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8323  027176  012712  052525                  MOV     #052525,(R2)    ;[DEST] = 052525
8324  027202  000257                          CCC                     ;SCOPE SYNC
8325  027204  000266                          266                     ;MAKE N:C=0110
8326
8327  027206  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8328
8329  027210  100003                          BPL     3$              ;N:C = 1000 ??
8330  027212  001402                          BEQ     3$
8331  027214  102401                          BVS     3$
8332  027216  103001                          BCC     4$
8333
8334  027220  104001                  3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
8335
8336  027222  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT?
8337  027224  001402                          BEQ     TST420          ;;BR IF YES
8338
8339  027226  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8340  027230  104001                  5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
8341
8342                                  ;;**************************************************************
8343                                  ;*TEST 420      "XOR RA,(RB)" TEST - A=052525,B=125252 N:C-1001
8344                                  ;;**************************************************************
8345  027232                          TST420:
8346  027232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8347  027234  012700  000420                  MOV     #420,R0         ;;LOAD R0 WITH TEST NUMBER
8348  027240  013701  027270                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8349  027244  012704  177777                  MOV     #-1,R4          ;RESULT S/B = 177777
8350  027250  012705  052525                  MOV     #52525,R5       ;MASK = 052525
8351  027254  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8352  027260  012712  125252                  MOV     #125252,(R2)    ;[DEST] = 125252
8353  027264  000257                          CCC                     ;SCOPE SYNC
8354  027266  000271                          271                     ;MAKE N:C=1001
8355
8356  027270  074512                  2$:     XOR     R5,(R2)         ;TEST THE XOR
8357
```

```
 8358  027272  100003                  BPL     3$              ;N:C = 1001 ??
 8359  027274  001402                  BEQ     3$
 8360  027276  102401                  BVS     3$
 8361  027300  103401                  BLS     4$
 8362
 8363  027302  104001          3$:     ERROR   1               ;XOR FAILED TO ALTER CODES PROPERLY
 8364
 8365  027304  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT?
 8366  027306  001402                  BEQ     TST421          ;;BR IF YES
 8367
 8368  027310  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 8369  027312  104001          5$:     ERROR   1               ;XOR DELIVERED THE WRONG RESULT
 8370
 8371
 8372                          ;;*****************************************************************
                               ;*TEST 421       SUB TEST SMO,DMO - (SRC) = (DEST) = +,+
 8373                          ;;*****************************************************************
 8374  027314                  TST421:
 8375  027314  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8376  027316  012700  000421          MOV     #421,R0         ;;LOAD R0 WITH TEST NUMBER
 8377  027322  013701  027342          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8378  027326  005004                  CLR     R4              ;RESULT S / B = 0
 8379  027330  012703  052525          MOV     #052525,R3      ;[R3] = DEST OP = 52525
 8380  027334  010305                  MOV     R3,R5           ;[R5] = SRC OP = 52525
 8381  027336  000257                  CCC                     ;CLEAR FLAGS
 8382  027340  000273                  273                     ;MAKE N:C = 1011
 8383
 8384  027342  160503          2$:     SUB     R5,R3           ;TEST THE SUB
 8385
 8386  027344  100403                  BMI     3$
 8387  027346  001002                  BNE     3$              ;DID N:C = 0100
 8388  027350  102401                  BVS     3$
 8389  027352  103001                  BCC     4$
 8390
 8391  027354  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
 8392
 8393  027356  020304          4$:     CMP     R3,R4           ;WAS RESULT = 0?
 8394  027360  001401                  BEQ     TST422          ;;BR IF YES
 8395
 8396  027362  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
 8397
 8398                          ;;*****************************************************************
                               ;*TEST 422       SUB TEST SMO,DMO - (SRC) = (DEST) = -,-
 8399                          ;;*****************************************************************
 8400
 8401  027364                  TST422:
 8402  027364  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8403  027366  012700  000422          MOV     #422,R0         ;;LOAD R0 WITH TEST NUMBER
 8404  027372  013701  027412          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8405  027376  005004                  CLR     R4              ;RESULT S / B = 0
 8406  027400  012703  125252          MOV     #125252,R3      ;[R3] = DEST OP = 125252
 8407  027404  010305                  MOV     R3,R5           ;[R5] = SOURCE OP = 125252
 8408  027406  000257                  CCC                     ;CLEAR FLAGS
 8409  027410  000273                  273                     ;MAKE N:C = 1011
 8410
 8411  027412  160503          2$:     SUB     R5,R3           ;TEST THE SUB
 8412
 8413  027414  100403                  BMI     3$
```

```
8414  027416  001002                  BNE     3$              ;N:C = 0100?
8415  027420  102401                  BVS     3$
8416  027422  103001                  BCC     4$
8417
8418  027424  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8419
8420  027426  020304          4$:     CMP     R3,R4           ;RESULT = 0?
8421  027430  001401                  BEQ     TST423          ;;BR IF YES
8422
8423  027432  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8424
8425                          ;;*****************************************************************
8426                          ;*TEST 423       SUB TEST SM0,DM0 - (SRC) = (DEST) = -,+
8427                          ;;*****************************************************************
8428  027434                  TST423:
8429  027434  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8430  027436  012700  000423          MOV     #423,R0         ;;LOAD R0 WITH TEST NUMBER
8431  027442  013701  027466          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8432  027446  012704  000002          MOV     #2,R4           ;RESULT S / B = 2
8433  027452  012703  000001          MOV     #1,R3           ;[R3] = DEST OP = 1
8434  027456  012705  177777          MOV     #-1,R5          ;[R5] = SRC OP = -1
8435  027462  000257                  CCC                     ;CLEAR FLAGS
8436  027464  000276                  276                     ;MAKE N:C = 1110
8437
8438  027466  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8439
8440  027470  100403                  BMI     3$
8441  027472  001402                  BEQ     3$              ;N:C - 0001
8442  027474  102401                  BVS     3$
8443  027476  103401                  BCS     4$
8444
8445  027500  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8446
8447  027502  020304          4$:     CMP     R3,R4           ;RESULT = +2?
8448  027504  001401                  BEQ     TST424          ;;BR IF YES
8449
8450  027506  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8451
8452                          ;;*****************************************************************
8453                          ;*TEST 424       SUB TEST SM0,DM0 (SRC) = -(DEST) = +,-
8454                          ;;*****************************************************************
8455  027510                  TST424:
8456  027510  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8457  027512  012700  000424          MOV     #424,R0         ;;LOAD R0 WITH TEST NUMBER
8458  027516  013701  027542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8459  027522  012704  177776          MOV     #-2,R4          ;RESULT S / B = -2
8460  027526  012703  177777          MOV     #-1,R3          ;[R3] = [DEST] = -1
8461  027532  012705  000001          MOV     #1,R5           ;[R5] = [SOURCE] = +1
8462  027536  000257                  CCC                     ;CLEAR FLAGS
8463  027540  000267                  267                     ;MAKE N:C = 0111
8464
8465  027542  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8466
8467  027544  100003                  BPL     3$
8468  027546  001402                  BEQ     3$              ;N:C = 1000
8469  027550  102401                  BVS     3$
```

```
8470  027552  103001                  BCC     4$
8471
8472  027554  104002          3$:     ERROR   2               ;SUB DID NOT ALTER CODES PROPERLY
8473
8474  027556  020403          4$:     CMP     R4,R3           ;RESULT = -2?
8475  027560  001401                  BEQ     TST425          ;;BR IF YES
8476
8477  027562  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8478
8479                          ;;*****************************************************************
8480                          ;*TEST 425       SUB TEST SM0,DM0 - "V" BIT SETS
8481                          ;;*****************************************************************
8482  027564                  TST425:
8483  027564  000004                 SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8484  027566  012700  000425         MOV     #425,R0         ;;LOAD R0 WITH TEST NUMBER
8485  027572  013701  027616         MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8486  027576  012704  077777         MOV     #77777,R4       ;RESULT = 77777
8487  027602  012703  100000         MOV     #100000,R3      ;[R3] = DEST OP = 100000
8488  027606  012705  000001         MOV     #1,R5           ;[R5] = SRC OP = 1
8489  027612  000257                 CCC                     ;CLEAR FLAGS
8490  027614  000274                 274                     ;MAKE N:C = 1100
8491
8492  027616  160503          2$:     SUB     R5,R3           ;TEST THE SUB
8493
8494  027620  100403                  BMI     3$
8495  027622  001402                  BEQ     3$              ;N:C = 0011 ("V" BIT SHOULD SET)
8496  027624  102001                  BVC     3$
8497  027626  103001                  BCC     4$
8498
8499  027630  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8500
8501  027632  020304          4$:     CMP     R3,R4           ;RESULT = 77777?
8502  027634  001401                  BEQ     TST426          ;;BR IF YES
8503
8504  027636  104002          5$:     ERROR   2               ;SUB DELIVERED WRONG RESULT
8505
8506                          ;;*****************************************************************
8507                          ;*TEST 426       SUB TEST - SM0,DM1 - N:C = 0110
8508                          ;;*****************************************************************
8509  027640                  TST426:
8510  027640  000004                 SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8511  027642  012700  000426         MOV     #426,R0         ;;LOAD R0 WITH TEST NUMBER
8512  027646  013701  027674         MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8513  027652  012702  063312         MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8514  027656  012704  177777         MOV     #-1,R4          ;RESULT S / B = 177777
8515  027662  012705  000001         MOV     #+1,R5          ;SRC OPR = +1
8516  027666  005012                 CLR     (R2)            ;[DEST] = 000000
8517  027670  000257                 CCC                     ;CLEAR FLAGS
8518  027672  000266                 266                     ;N:C = 0110
8519
8520  027674  160512          2$:     SUB     R5,(R2)         ;TEST THE SUB
8521
8522  027676  100003                  BPL     3$              ;N:C = 1001
8523  027700  001402                  BEQ     3$
8524  027702  102401                  BVS     3$
8525  027704  103401                  BCS     4$
```

```
8526
8527  027706  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8528
8529  027710  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8530  027712  001402                          BEQ     TST427          ;;BR IF YES
8531
8532  027714  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8533  027716  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8534
8535                                  ;;**********************************************************
8536                                  ;*TEST 427       SUB TEST - SM0,DM1 - N:C = 1010
8537                                  ;;**********************************************************
8538  027720                          TST427:
8539  027720  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8540  027722  012700  000427                  MOV     #427,R0         ;;LOAD R0 WITH TEST NUMBER
8541  027726  013701  027754                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8542  027732  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8543  027736  005004                          CLR     R4              ;RESULT S / B = 000000
8544  027740  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
8545  027744  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
8546  027750  000257                          CCC                     ;CLEAR FLAGS
8547  027752  000272                          272                     ;N:C = 1010
8548
8549  027754  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8550
8551  027756  100403                          BMI     3$              ;N:C = 0100
8552  027760  001002                          BNE     3$
8553  027762  102401                          BVS     3$
8554  027764  103001                          BCC     4$
8555
8556  027766  104001                  3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8557
8558  027770  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8559  027772  001402                          BEQ     TST430          ;;BR IF YES
8560
8561  027774  011203                          MOV     (R2),R3         ;GET THE WAS DATA
8562  027776  104001                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8563
8564                                  ;;**********************************************************
8565                                  ;*TEST 430       SUB TEST - SM0,DM1 - N:C = 0000
8566                                  ;;**********************************************************
8567  030000                          TST430:
8568  030000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8569  030002  012700  000430                  MOV     #430,R0         ;;LOAD R0 WITH TEST NUMBER
8570  030006  013701  030034                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8571  030012  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8572  030016  012704  077777                  MOV     #77777,R4       ;RESULT S / B = 77777
8573  030022  012705  000001                  MOV     #+1,R5          ;SRC OPR = +1
8574  030026  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
8575  030032  000257                          CCC                     ;CLEAR FLAGS
8576
8577  030034  160512                  2$:     SUB     R5,(R2)         ;TEST THE SUB
8578
8579  030036  100403                          BMI     3$              ;N:C = 0010
8580  030040  001402                          BEQ     3$
8581  030042  102001                          BVC     3$
```

```
8582  030044  103001                  BCC     4$
8583
8584  030046  104001          3$:     ERROR   1               ;SUB FAILED TO ALTER CODES PROPERLY
8585
8586  030050  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
8587  030052  001402                  BEQ     TST431          ;;BR IF YES
8588
8589  030054  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8590  030056  104001          5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT
8591
8592                          ;;**************************************************************
8593                          ;*TEST 431       SUB TEST - SM1,DM0 - N:C = 0110
8594                          ;;**************************************************************
8595  030060                  TST431:
8596  030060  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8597  030062  012700  000431          MOV     #431,R0         ;;LOAD R0 WITH TEST NUMBER
8598  030066  013701  030110          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8599  030072  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
8600  030076  012705  064034          MOV     #DWTB+2,R5      ;SRC ADDR = DWTB+2
8601  030102  005003                  CLR     R3              ;[DEST] = 000000
8602  030104  000257                  CCC                     ;CLEAR FLAGS
8603  030106  000266                  266                     ;N:C = 0110
8604
8605  030110  161503          2$:     SUB     (R5),R3         ;TEST THE SUB
8606
8607  030112  100003                  BPL     3$              ;N:C = 1001
8608  030114  001402                  BEQ     3$
8609  030116  102401                  BVS     3$
8610  030120  103401                  BCS     4$
8611
8612  030122  104002          3$:     ERROR   2               ;SUB FAILED TO ALTER CODES PROPERLY
8613
8614  030124  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
8615  030126  001401                  BEQ     TST432          ;;BR IF YES
8616
8617  030130  104002          5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT
8618
8619                          ;;**************************************************************
8620                          ;*TEST 432       SUB TEST - SM1,DM0 - N:C = 1010
8621                          ;;**************************************************************
8622  030132                  TST432:
8623  030132  000004                  SCOPE                   ; ALL THE SCOPE LOOP UTILITY
8624  030134  012700  000432          MOV     #432,R0         ;;LOAD R0 WITH TEST NUMBER
8625  030140  013701  030160          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8626  030144  005004                  CLR     R4              ;RESULT S / B = 000000
8627  030146  012705  063324          MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
8628  030152  011503                  MOV     (R5),R3         ;[DEST] = 177777
8629  030154  000257                  CCC                     ;CLEAR FLAGS
8630  030156  000272                  272                     ;N:C = 1010
8631
8632  030160  161503          2$:     SUB     (R5),R3 ;TEST THE SUB
8633
8634  030162  100403                  BMI     3$              ;N:C - 0100
8635  030164  001002                  BNE     3$
8636  030166  102401                  BVS     3$
8637  030170  103001                  BCC     4$
```

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

                                4$:     CMP     R4,R3           ;CORRECT RESULT ?
                                        BEQ     TST433          ;;BR IF YES

                                5$:     ERROR   2               ;SUB DELIVERED THE WRONG RESULT

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

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

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

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

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

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

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

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

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

```
                                                ;;*****************************************************************
                                                ;*TEST 435       SUB  SM1,DM2 TEST - N:C = 0110
                                                ;;*****************************************************************
030344                                          TST435:
030344  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030346  012700  000435                                  MOV     #435,R0         ;;LOAD R0 WITH TEST NUMBER
030352  013701  030406                                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030356  012702  063312                                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030362  012704  177777                                  MOV     #-1,R4          ;RESULT S / B = 177777
030366  012705  063316                                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
030372  012715  000001                                  MOV     #+1,(R5)        ;[SOURCE] = 000001
030376  005012                                          CLR     (R2)            ;[DEST] = 000000
030400  010203                                          MOV     R2,R3           ;R3 GETS DEST ADDR
030402  000257                                          CCC                     ;CLEAR FLAGS
030404  000266                                          266                     ;N:C = 0110

030406  161523                                  2$:     SUB     (R5),(R3)+      ;TEST THE SUB

030410  100003                                          BPL     3$              ;N:C = 1001 ?
030412  001402                                          BEQ     3$
030414  102401                                          BVS     3$
030416  103401                                          BCS     4$

030420  104005                                  3$:     ERROR   5               ;SUB FAILED TO ALTER CODES PROPERLY

030422  020412                                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
030424  001402                                          BEQ     TST436          ;;BR IF YES

030426  011203                                          MOV     (R2),R3         ;GET THE WAS DATA
030430  104001                                  5$:     ERROR   1               ;SUB DELIVERED THE WRONG RESULT

                                                ;;*****************************************************************
                                                ;*TEST 436       NEG DM2 TEST
                                                ;;*****************************************************************
030432                                          TST436:
030432  000004                                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
030434  012700  000436                                  MOV     #436,R0         ;;LOAD R0 WITH TEST NUMBER
030440  013701  030464                                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
030444  012702  063312                                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
030450  012704  125252                                  MOV     #125252,R4      ;RESULT S / B = 125252
030454  010205                                          MOV     R2,R5           ;[R5] = DEST ADDR
030456  012712  052526                                  MOV     #52526,(R2)     ;[DEST] = 52526
030462  000257                                          CCC                     ;SCOPE SYNC

030464  005425                                  2$:     NEG     (R5)+           ;TEST THE NEG - MODE 2

030466  020412                                          CMP     R4,(R2)         ;RESULT = 125252?
```

```
8750  030470  001402                  BEQ     4$              ;BR IF YES
8751
8752  030472  011203                  MOV     (R2),R3         ;GET THE WAS DATA
8753  030474  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8754
8755  030476  022705  063314  4$:     CMP     #MBUF0+2,R5     ;DID REG. GET AUTO INCREMENTED?
8756  030502  001401                  BEQ     TST437          ;;BR IF YES
8757
8758  030504  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8759
8760                          ;;*****************************************************************
8761                          ;*TEST 437       NEG DM3 TEST
8762                          ;;*****************************************************************
8763  030506                  TST437:
8764  030506  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8765  030510  012700  000437          MOV     #437,R0         ;;LOAD R0 WITH TEST NUMBER
8766  030514  013701  030542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8767  030520  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8768  030524  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8769  030530  012705  063306          MOV     #ATA+10,R5      ;[ATA+10] = MBUF0
8770  030534  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8771  030540  000257                  CCC                     ;SCOPE SYNC
8772
8773  030542  005435          2$:     NEG     @(R5)+          ;TEST THE NEG - MODE 3
8774
8775  030544  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8776  030546  001402                  BEQ     4$              ;BR IF YES
8777
8778  030550  011203                  MOV     (R2),R3         ;GET WAS DATA
8779  030552  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
8780
8781  030554  022705  063310  4$:     CMP     #ATA+12,R5      ;DID REG GET AUTO INCREMENTED?
8782  030560  001401                  BEQ     TST440          ;;BR IF YES
8783
8784  030562  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG.
8785
8786                          ;;*****************************************************************
8787                          ;*TEST 440       NEG DM4 TEST
8788                          ;;*****************************************************************
8789  030564                  TST440:
8790  030564  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8791  030566  012700  000440          MOV     #440,R0         ;;LOAD R0 WITH TEST NUMBER
8792  030572  013701  030620          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8793  030576  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8794  030602  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8795  030606  012705  063314          MOV     #MBUF0+2,R5     ;[R5] = DEST ADDR + 2
8796  030612  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8797  030616  000257                  CCC                     ;SCOPE SYNC
8798
8799  030620  005445          2$:     NEG     -(R5)           ;TEST THE NEG - MODE 4
8800
8801  030622  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8802  030624  001402                  BEQ     4$              ;BR IF YES
8803
8804  030626  011203                  MOV     (R2),R3         ;GET WAS DATA
8805  030630  104001          3$:     ERROR   1               ;NEG DELIVERED WRONG RESULT
```

```
 8806
 8807  030632  020502          4$:     CMP     R5,R2           ;DID REG GET AUTO INCREMENTED?
 8808  030634  001401                  BEQ     TST441          ;;BR IF YES
 8809
 8810  030636  104005          5$:     ERROR   5               ;NEG FAILED TO UPDATE REG
```

```
                                ;;*****************************************************************
                                ;*TEST 441       NEG DM5 TEST
                                ;;*****************************************************************
030640                          TST4 1:
                                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
030640  000004
030642  012700  000441                  MOV     #441,R0                 ;;LOAD R0 WITH TEST NUMBER
030646  013701  030674                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
030652  012702  063312                  MOV     #MBUF0,R2               ;DEST ADDR = MBUF0
030656  012704  125252                  MOV     #125252,R4              ;RESULT S / B = 125252
030662  012705  063310                  MOV     #ATA+12,R5              ;[R5] = (ADR OF MBUF0) +2
030666  012712  052526                  MOV     #52526,(R2)             ;[DEST] = 52526
030672  000257                        CCC                             ;SCOPE SYNC
                                
030674  005455                  2$:     NEG     @-(R5)                  ;TEST THE NEG - MODE 5
                                
030676  020412                        CMP     R4,(R2)                 ;RESULT = 125252?
030700  001402                        BEQ     4$                      ;BR IF YES
                                
030702  011203                        MOV     (R2),R3                 ;GET WAS DATA
030704  104001                  3$:     ERROR   1                       ;NEG DELIVERED WRONG RESULT
                                
030706  022705  063306          4$:     CMP     #ATA+10,R5              ;DID NEG UPDATE REG
030712  001401                        BEQ     TST442                  ;;BR IF YES
                                
030714  104005                  5$:     ERROR   5                       ;NEG FAILED TO UPDATE REG
                                
                                ;;*****************************************************************
                                ;*TEST 442       NEG DM6 TEST
                                ;;*****************************************************************
030716                          TST442:
                                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
030716  000004
030720  012700  000442                  MOV     #442,R0                 ;;LOAD R0 WITH TEST NUMBER
030724  013701  030752                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
030730  012702  063312                  MOV     #MBUF0,R2               ;DEST ADDR = MBUF0
030734  012704  125252                  MOV     #125252,R4              ;RESULT S / B = 125252
030740  012705  063310                  MOV     #MBUF0-2,R5             ;[R5] = BASE ADDR
030744  012712  052526                  MOV     #52526,(R2)             ;[DEST] = 52526
030750  000257                        CCC                             ;SCOPE SYNC
                                
030752  005465  000002          2$:     NEG     2(R5)                   ;TEST THE NEG - MODE 6
                                
030756  020412                        CMP     R4,(R2)                 ;RESULT = 125252?
030760  001402                        BEQ     TST443                  ;;BR IF YES
                                
030762  011203                        MOV     (R2),R3                 ;GET WAS DATA
030764  104001                  3$:     ERROR   1                       ;NEG DELIVERED WRONG RESULT
                                
                                ;;*****************************************************************
                                ;*TEST 443       NEG DM7 TEST
                                ;;*****************************************************************
030766                          TST443:
                                        SCOPE                           ;CALL THE SCOPE LOOP UTILITY
030766  000004
030770  012700  000443                  MOV     #443,R0                 ;;LOAD R0 WITH TEST NUMBER
030774  013701  031022                  MOV     @#2$,R1                 ;LOAD R1 WITH TEST INSTRUCTION WORD
031000  012702  063312                  MOV     #MBUF0,R2               ;DEST ADDR   MBUF0
```

```
8867  031004  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
8868  031010  012705  063276          MOV     #ATA,R5         ;[R5] = BASE ADDR
8869  031014  012712  052526          MOV     #52526,(R2)     ;[DEST] = 52526
8870  031020  000257                  CCC                     ;SCOPE SYNC
8871
8872  031022  005475  000010  2$:     NEG     @10(R5)         ;TEST THE NEG - MODE 7
8873
8874  031026  020412                  CMP     R4,(R2)         ;RESULT = 125252?
8875  031030  001402                  BEQ     TST444          ;;BR IF YES
8876
8877  031032  011203                  MOV     (R2),R3         ;GET WAS DATA
8878  031034  104001  3$:     ERROR   1                       ;NEG DELIVERED WRONG RESULT
8879
8880                          ;;**************************************************************
8881                          ;*TEST 444       MOV SM1,DM1 TEST - N:C = 0100
8882                          ;;**************************************************************
8883  031036                  TST444:
8884  031036  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8885  031040  012700  000444          MOV     #444,R0         ;;LOAD R0 WITH TEST NUMBER
8886  031044  013701  031074          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8887  031050  005004                  CLR     R4              ;RESULT S / B = 177777
8888  031052  005104                  COM     R4
8889  031054  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8890  031060  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
8891  031064  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
8892  031066  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
8893  031070  000257                  CCC                     ;CLEAR FLAGS
8894  031072  000264                  264                     ;N:C = 0100
8895
8896  031074  011513          2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
8897
8898  031076  100003                  BPL     3$              ;N:C = 1000 ?
8899  031100  001402                  BEQ     3$
8900  031102  102401                  BVS     3$
8901  031104  103001                  BCC     4$
8902
8903  031106  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
8904
8905  031110  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
8906  031112  001403                  BEQ     TST445          ;;BR IF YES
8907
8908  031114  005003                  CLR     R3              ;GET THE WAS DATA
8909  031116  051203                  BIS     (R2),R3
8910  031120  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
8911
8912                          ;;**************************************************************
8913                          ;*TEST 445       MOV SM2,DM1 TEST - N:C = 0100
8914                          ;;**************************************************************
8915  031122                  TST445:
8916  031122  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
8917  031124  012700  000445          MOV     #445,R0         ;;LOAD R0 WITH TEST NUMBER
8918  031130  013701  031160          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
8919  031134  005004                  CLR     R4              ;RESULT S / B = 177777
8920  031136  005104                  COM     R4
8921  031140  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
8922  031144  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
```

```
 8923  031150  010203                  MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8924  031152  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 8925  031154  000257                  CCC                     ;CLEAR FLAGS
 8926  031156  000264                  264                     ;N:C = 0100
 8927
 8928  031160  012513          2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
 8929
 8930  031162  100003                  BPL     3$              ;N:C = 1000 ?
 8931  031164  001402                  BEQ     3$
 8932  031166  102401                  BVS     3$
 8933  031170  103001                  BCC     4$
 8934
 8935  031172  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8936
 8937  031174  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 8938  031176  001403                  BEQ     TST446          ;;BR IF YES
 8939
 8940  031200  005003                  CLR     R3              ;GET THE WAS DATA
 8941  031202  051203                  BIS     (R2),R3
 8942  031204  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8943
 8944                          ;;*****************************************************************
 8945                          ;*TEST 446       MOV SM1,DM1 TEST - N:C = 1011
 8946                          ;;*****************************************************************
 8947  031206                  TST446:
 8948  031206  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8949  031210  012700  000446          MOV     #446,R0         ;;LOAD R0 WITH TEST NUMBER
 8950  031214  013701  031244          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8951  031220  005004                  CLR     R4              ;RESULT S / B = 000000
 8952  031222  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8953  031226  012705  063322          MOV     #DWTA,R5        ;SOURCE ADDR - DWTA
 8954  031232  010203                  MOV     R2,R3           ;BASE DEST ADDR - MBUF0
 8955  031234  005012                  CLR     (R2)            ;MAKE [DEST] = 177777
 8956  031236  005112                  COM     (R2)
 8957  031240  000257                  CCC                     ;CLEAR FLAGS
 8958  031242  000273                  273                     ;N:C = 1011
 8959
 8960  031244  011513          2$:     MOV     (R5),(R3)       ;TEST THE MOV - SM1,DM1
 8961
 8962  031246  100403                  BMI     3$              ;N:C - 0101 ?
 8963  031250  001002                  BNE     3$
 8964  031252  102401                  BVS     3$
 8965  031254  103401                  BCS     4$
 8966
 8967  031256  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 8968
 8969  031260  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 8970  031262  001403                  BEQ     TST447          ;;BR IF YES
 8971
 8972  031264  005003                  CLR     R3              ;GET THE WAS DATA
 8973  031266  051203                  BIS     (R2),R3
 8974  031270  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 8975
 8976                          ;;*****************************************************************
 8977                          ;*TEST 447       MOV SM2,DM1 TEST - N:C = 1011
 8978                          ;;*****************************************************************
```

```
 8979  031272
 8980  031272  000004                  TST447:
 8981  031274  012700  000447                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 8982  031300  013701  031330                  MOV     #447,R0         ;;LOAD R0 WITH TEST NUMBER
 8983  031304  005004                          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 8984  031306  012702  063312                  CLR     R4              ;RESULT S / B = 000000
 8985  031312  012705  063322                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 8986  031316  010203                          MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
 8987  031320  005012                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 8988  031322  005112                          CLR     (R2)            ;MAKE [DEST] = 177777
 8989  031324  000257                          COM     (R2)
 8990  031326  000273                          CCC                     ;CLEAR FLAGS
 8991                                          273                     ;N:C = 1011
 8992  031330  012513                  2$:     MOV     (R5)+,(R3)      ;TEST THE MOV - SM2,DM1
 8993
 8994  031332  100403                          BMI     3$              ;N:C = 0101 ?
 8995  031334  001002                          BNE     3$
 8996  031336  102401                          BVS     3$
 8997  031340  103401                          BCS     4$
 8998
 8999  031342  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9000
 9001  031344  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9002  031346  001403                          BEQ     TST450          ;;BR IF YES
 9003
 9004  031350  005003                          CLR     R3              ;GET THE WAS DATA
 9005  031352  051203                          BIS     (R2),R3
 9006  031354  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9007
 9008                                  ;;*********************************************************************
 9009                                  ;*TEST 450       MOV SM1,DM2 TEST - N:C = 0100
 9010                                  ;;*********************************************************************
 9011  031356                          TST450:
 9012  031356  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9013  031360  012700  000450                  MOV     #450,R0         ;;LOAD R0 WITH TEST NUMBER
 9014  031364  013701  031414                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9015  031370  005004                          CLR     R4              ;RESULT S / B = 177777
 9016  031372  005104                          COM     R4
 9017  031374  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9018  031400  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
 9019  031404  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9020  031406  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9021  031410  000257                          CCC                     ;CLEAR FLAGS
 9022  031412  000264                          264                     ;N:C = 0100
 9023
 9024  031414  011523                  2$:     MOV     (R5),(R3)+      ;TEST THE MOV - SM1,DM2
 9025
 9026  031416  100003                          BPL     3$              ;N:C =  1000 ?
 9027  031420  001402                          BEQ     3$
 9028  031422  102401                          BVS     3$
 9029  031424  103001                          BCC     4$
 9030
 9031  031426  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9032
 9033  031430  022703  063314          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
 9034  031434  001401                          BEQ     6$              ;BR IF YES
```

```
 9035
 9036  031436  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9037
 9038  031440  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9039  031442  001403                          BEQ     TST451          ;;BR IF YES
 9040
 9041  031444  005003                          CLR     R3              ;GET THE WAS DATA
 9042  031446  051203                          BIS     (R2),R3
 9043  031450  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9044
 9045                                  ;;*********************************************************************
 9046                                  ;*TEST 451       MOV SM2,DM2 TEST - N:C = 0100
 9047                                  ;;*********************************************************************
 9048  031452                          TST451:
 9049  031452  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9050  031454  012700  000451                  MOV     #451,R0         ;;LOAD R0 WITH TEST NUMBER
 9051  031460  013701  031510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9052  031464  005004                          CLR     R4              ;RESULT S / B = 177777
 9053  031466  005104                          COM     R4
 9054  031470  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9055  031474  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA
 9056  031500  010203                          MOV     R2,R3           ;BASE DEST ADDR = MBUF0
 9057  031502  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9058  031504  000257                          CCC                     ;CLEAR FLAGS
 9059  031506  000264                          264                     ;N:C = 0100
 9060
 9061  031510  012523                  2$:     MOV     (R5)+,(R3)+     ;TEST THE MOV - SM2,DM2
 9062
 9063  031512  100003                          BPL     3$              ;N:C = 1000 ?
 9064  031514  001402                          BEQ     3$
 9065  031516  102401                          BVS     3$
 9066  031520  103001                          BCC     4$
 9067
 9068  031522  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9069
 9070  031524  022703  06331+          4$:     CMP     #MBUF0+2,R3     ;DID MOV INCREMENT DEST REG ?
 9071  031530  001401                          BEQ     6$              ;BR IF YES
 9072
 9073  031532  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9074
 9075  031534  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9076  031536  001403                          BEQ     TST452          ;;BR IF YES
 9077
 9078  031540  005003                          CLR     R3              ;GET THE WAS DATA
 9079  031542  051203                          BIS     (R2),R3
 9080  031544  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9081
 9082                                  ;;*********************************************************************
 9083                                  ;*TEST 452       MOV SM1,DM3 TEST - N:C = 0100
 9084                                  ;;*********************************************************************
 9085  031546                          TST452:
 9086  031546  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9087  031550  012700  000452                  MOV     #452,R0         ;;LOAD R0 WITH TEST NUMBER
 9088  031554  013701  031606                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9089  031560  005004                          CLR     R4              ;RESULT S / B = 177777
 9090  031562  005104                          COM     R4
```

```
 9091  031564  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9092  031570  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9093  031574  012703  063306                  MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9094  031600  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9095  031602  000257                          CCC                     ;CLEAR FLAGS
 9096  031604  000264                          264                     ;N:C = 0100
 9097
 9098  031606  011533                  2$:     MOV     (R5),@(R3)+     ;TEST THE MOV - SM1,DM3
 9099
 9100  031610  100003                          BPL     3$              ;N:C = 1000 ?
 9101  031612  001402                          BEQ     3$
 9102  031614  102401                          BVS     3$
 9103  031616  103001                          BCC     4$
 9104
 9105  031620  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9106
 9107  031622  022703  063310          4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9108  031626  001401                          BEQ     6$              ;BR IF YES
 9109
 9110  031630  104005                  5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9111
 9112  031632  020412                  6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9113  031634  001403                          BEQ     TST453          ;;BR IF YES
 9114
 9115  031636  005003                          CLR     R3              ;GET THE WAS DATA
 9116  031640  051203                          BIS     (R2),R3
 9117  031642  104001                  7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9118
 9119                                  ;;*********************************************************
 9120                                  ;*TEST 453       MOV SM2,DM3 TEST - N:C = 0100
 9121                                  ;*********************************************************
 9122  031644                          TST453:
 9123  031644  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9124  031646  012700  000453                  MOV     #453,R0         ;;LOAD R0 WITH TEST NUMBER
 9125  031652  013701  031704                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9126  031656  005004                          CLR     R4              ;RESULT S / B = 177777
 9127  031660  005104                          COM     R4
 9128  031662  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9129  031666  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9130  031672  012703  063306                  MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
 9131  031676  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9132  031700  000257                          CCC                     ;CLEAR FLAGS
 9133  031702  000264                          264                     ;N:C = 0100
 9134
 9135  031704  012533                  2$:     MOV     (R5)+,@(R3)+    ;TEST THE MOV - SM2,DM3
 9136
 9137  031706  100003                          BPL     3$              ;N:C = 1000 ?
 9138  031710  001402                          BEQ     3$
 9139  031712  102401                          BVS     3$
 9140  031714  103001                          BCC     4$
 9141
 9142  031716  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9143
 9144  031720  022703  063310          4$:     CMP     #ATA+12,R3      ;DID MOV INCREMENT DEST REG ?
 9145  031724  001401                          BEQ     6$              ;BR IF YES
 9146
```

```
 9147  031726  104005                 5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9148
 9149  031730  020412                 6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9150  031732  001403                         BEQ     TST454          ;;BR IF YES
 9151
 9152  031734  005003                         CLR     R3              ;GET THE WAS DATA
 9153  031736  051203                         BIS     (R2),R3
 9154  031740  104001                 7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9155
 9156                                 ;;*****************************************************************
 9157                                 ;*TEST 454       MOV SM1,DM4 TEST - N:C = 0100
 9158                                 ;;*****************************************************************
 9159  031742                         TST454:
 9160  031742  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9161  031744  012700  000454                 MOV     #454,R0         ;;LOAD R0 WITH TEST NUMBER
 9162  031750  013701  032002                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9163  031754  005004                         CLR     R4              ;RESULT S / B = 177777
 9164  031756  005104                         COM     R4
 9165  031760  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9166  031764  012705  063324                 MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9167  031770  012703  063314                 MOV     #MBUF0+2,R3     ;BASE DEST ADDR = MBUF0+2
 9168  031774  005012                         CLR     (R2)            ;MAKE [DEST] = 000000
 9169  031776  000257                         CCC                     ;CLEAR FLAGS
 9170  032000  000264                         264                     ;N:C = 0100
 9171
 9172  032002  011543                 2$:     MOV     (R5),-(R3)      ;TEST THE MOV - SM1,DM4
 9173
 9174  032004  100003                         BPL     3$              ;N:C   1000 ?
 9175  032006  001402                         BEQ     3$
 9176  032010  102401                         BVS     3$
 9177  032012  103001                         BCC     4$
 9178
 9179  032014  104001                 3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9180
 9181  032016  020203                 4$:     CMP     R2,R3           ;DID MOV DECREMENT DEST REG ?
 9182  032020  001401                         BEQ     6$              ;BR IF YES
 9183
 9184  032022  104005                 5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9185
 9186  032024  020412                 6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9187  032026  001403                         BEQ     TST455          ;;BR IF YES
 9188
 9189  032030  005003                         CLR     R3              ;GET THE WAS DATA
 9190  032032  051203                         BIS     (R2),R3
 9191  032034  104001                 7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9192
 9193                                 ;;*****************************************************************
 9194                                 ;*TEST 455       MOV SM2,DM4 TEST - N:C = 0100
 9195                                 ;;*****************************************************************
 9196  032036                         TST455:
 9197  032036  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9198  032040  012700  000455                 MOV     #455,R0         ;;LOAD R0 WITH TEST NUMBER
 9199  032044  013701  032076                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9200  032050  005004                         CLR     R4              ;RESULT S / B = 177777
 9201  032052  005104                         COM     R4
 9202  032054  012702  063312                 MOV     #MBUF0,R2       ;DEST ADDR - MBUF0
```

```
 9203  032060  012705  063324              MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9204  032064  012703  063314              MOV     #MBUF0+2,R3             ;BASE DEST ADDR = MBUF0+2
 9205  032070  005012                      CLR     (R2)            ;MAKE [DEST] = 000000
 9206  032072  000257                      CCC                     ;CLEAR FLAGS
 9207  032074  000264                      264                     ;N:C = 0100
 9208
 9209  032076  012543              2$:     MOV     (R5)+,-(R3)     ;TEST THE MOV - SM2,DM4
 9210
 9211  032100  100003                      BPL     3$              ;N:C - 1000 ?
 9212  032102  001402                      BEQ     3$
 9213  032104  102401                      BVS     3$
 9214  032106  103001                      BCC     4$
 9215
 9216  032110  104001              3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9217
 9218  032112  020203              4$:     CMP     R2,R3           ;DID MOV INCREMENT DEST REG ?
 9219  032114  001401                      BEQ     6$              ;BR IF YES
 9220
 9221  032116  104005              5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9222
 9223  032120  020412              6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9224  032122  001403                      BEQ     TST456          ;;BR IF YES
 9225
 9226  032124  005003                      CLR     R3              ;GET THE WAS DATA
 9227  032126  051203                      BIS     (R2),R3
 9228  032130  104001              7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9229
 9230                              ;;*****************************************************************
 9231                              ;*TEST 456      MOV SM1,DM5 TEST - N:C - 0100
 9232                              ;;*****************************************************************
 9233  032132                      TST456:
 9234  032132  000004                      SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9235  032134  012700  000456              MOV     #456,R0         ;;LOAD R0 WITH TEST NUMBER
 9236  032140  013701  032204              MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9237                              .SBTTL USER CONTROLLED BREAKPOINT -- BIT9
 9238  032144  032737  001000  063234      BIT     #BIT9,@#BPTLOC  ;BREAKPOINT HALT SET ??
 9239  032152  001401                      BEQ     .+4             ;BR IF NOT
 9240  032154  000000                      HALT                    ;BREAK - DEPRESS CONTINUE TO RESTART
 9241  032156  005004                      CLR     R4              ;RESULT S / B = 177777
 9242  032160  005104                      COM     R4
 9243  032162  012702  063312              MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9244  032166  012705  063324              MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9245  032172  012703  063310              MOV     #ATA+12,R3              ;BASE DEST ADDR = ATA+12
 9246  032176  005012                      CLR     (R2)            ;MAKE [DEST] = 000000
 9247  032200  000257                      CCC                     ;CLEAR FLAGS
 9248  032202  000264                      264                     ;N:C = 0100
 9249
 9250  032204  011553              2$:     MOV     (R5),@-(R3)     ;TEST THE MOV - SM1,DM5
 9251
 9252  032206  100003                      BPL     3$              ;N:C = 0100 ?
 9253  032210  001402                      BEQ     3$
 9254  032212  102401                      BVS     3$
 9255  032214  103001                      BCC     4$
 9256
 9257  032216  104001              3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9258
```

```
 9259  032220  022703  063306        4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9260  032224  001401                        BEQ     6$              ;BR IF YES
 9261
 9262  032226  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9263
 9264  032230  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9265  032232  001403                        BEQ     TST457          ;;BR IF YES
 9266
 9267  032234  005003                        CLR     R3              ;GET THE WAS DATA
 9268  032236  051203                        BIS     (R2),R3
 9269  032240  104001                7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9270
 9271                                ;;*****************************************************************
 9272                                ;*TEST 457       MOV SM2,DM5 TEST - N:C = 0100
 9273                                ;;*****************************************************************
 9274  032242                        TST457:
 9275  032242  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9276  032244  012700  000457                MOV     #457,R0         ;;LOAD R0 WITH TEST NUMBER
 9277  032250  013701  032302                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9278  032254  005004                        CLR     R4              ;RESULT S / B = 177777
 9279  032256  005104                        COM     R4
 9280  032260  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9281  032264  012705  063324                MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9282  032270  012703  063310                MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
 9283  032274  005012                        CLR     (R2)            ;MAKE [DEST] = 00000
 9284  032276  000257                        CCC                     ;CLEAR FLAGS
 9285  032300  000264                        264                     ;N:C = 1000
 9286
 9287  032302  012553                2$:     MOV     (R5)+,@-(R3)    ;TEST THE MOV - SM2,DM5
 9288
 9289  032304  100003                        BPL     3$              ;N:C = 1000 ?
 9290  032306  001402                        BEQ     3$
 9291  032310  102401                        BVS     3$
 9292  032312  103001                        BCC     4$
 9293
 9294  032314  104001                3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9295
 9296  032316  022703  063306        4$:     CMP     #ATA+10,R3      ;DID MOV DECREMENT DEST REG ?
 9297  032322  001401                        BEQ     6$              ;BR IF YES
 9298
 9299  032324  104005                5$:     ERROR   5               ;MOV FAILED TO UPDATE DEST REG
 9300
 9301  032326  020412                6$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9302  032330  001403                        BEQ     TST460          ;;BR IF YES
 9303
 9304  032332  005003                        CLR     R3              ;GET THE WAS DATA
 9305  032334  051203                        BIS     (R2),R3
 9306  032336  104001                7$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9307
 9308                                ;;*****************************************************************
 9309                                ;*TEST 460       MOV SM1,DM6 TEST - N:C - 0100
 9310                                ;;*****************************************************************
 9311  032340                        TST460:
 9312  032340  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9313  032342  012700  000460                MOV     #460,R0         ;;LOAD R0 WITH TEST NUMBER
 9314  032346  013701  032400                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
```

```
 9315  032352  005004                          CLR     R4              ;RESULT S / B = 177777
 9316  032354  005104                          COM     R4
 9317  032356  012702  063320                  MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9318  032362  012705  063324                  MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9319  032366  012703  063312                  MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9320  032372  005012                          CLR     (R2)            ;MAKE [DEST] = 000000
 9321  032374  000257                          CCC                     ;CLEAR FLAGS
 9322  032376  000264                          264                     ;N:C = 0100
 9323
 9324  032400  011563  000006          2$:     MOV     (R5),6(R3)      ;TEST THE MOV - SM1,DM6
 9325
 9326  032404  100003                          BPL     3$              ;N:C = 1000 ?
 9327  032406  001402                          BEQ     3$
 9328  032410  102401                          BVS     3$
 9329  032412  103001                          BCC     4$
 9330
 9331  032414  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9332
 9333  032416  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9334  032420  001403                  BEQ     TST461          ;;BR IF YES
 9335
 9336  032422  005003                  CLR     R3              ;GET THE WAS DATA
 9337  032424  051203                  BIS     (R2),R3
 9338  032426  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9339
 9340                          ;;*****************************************************************
 9341                          ;*TEST 461      MOV SM2,DM6 TEST - N:C = 0100
 9342                          ;;*****************************************************************
 9343  032430                  TST461:
 9344  032430  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9345  032432  012700  000461          MOV     #461,R0         ;;LOAD R0 WITH TEST NUMBER
 9346  032436  013701  032470          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9347  032442  005004                  CLR     R4              ;RESULT S / B = 177777
 9348  032444  005104                  COM     R4
 9349  032446  012702  063320          MOV     #MBUF0+6,R2     ;DEST ADDR = MBUF0+6
 9350  032452  012705  063324          MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
 9351  032456  012703  063312          MOV     #MBUF0,R3       ;BASE DEST ADDR = MBUF0
 9352  032462  005012                  CLR     (R2)            ;MAKE [DEST] = 000000
 9353  032464  000257                  CCC                     ;CLEAR FLAGS
 9354  032466  000264                  264                     ;N:C = 0100
 9355
 9356  032470  012563  000006  2$:     MOV     (R5)+,6(R3)     ;TEST THE MOV - SM2,DM6
 9357
 9358  032474  100003                  BPL     3$              ;N:C = 1000 ?
 9359  032476  001402                  BEQ     3$
 9360  032500  102401                  BVS     3$
 9361  032502  103001                  BCC     4$
 9362
 9363  032504  104001          3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9364
 9365  032506  020412          4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9366  032510  001403                  BEQ     TST462          ;;BR IF YES
 9367
 9368  032512  005003                  CLR     R3              ;GET THE WAS DATA
 9369  032514  051203                  BIS     (R2),R3
 9370  032516  104001          5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
```

```
                        ;;*****************************************************************
                        ;*TEST 462       MOV SM1,DM7 TEST - N:C = 0100
                        ;;*****************************************************************
032520                  TST462:
032520  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032522  012700  000462            MOV     #462,R0         ;;LOAD R0 WITH TEST NUMBER
032526  013701  032560            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032532  005004                    CLR     R4              ;RESULT S / B = 177777
032534  005104                    COM     R4
032536  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032542  012705  063324            MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032546  012703  063276            MOV     #ATA,R3         ;BASE DEST ADDR = ATA
032552  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
032554  000257                    CCC                     ;CLEAR FLAGS
032556  000264                    264                     ;N:C = 0100
                        
032560  011573  000010    2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM1,DM7
                        
032564  100003                    BPL     3$              ;N:C = 1000 ?
032566  001402                    BEQ     3$
032570  102401                    BVS     3$
032572  103001                    BCC     4$
                        
032574  104001            3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
                        
032576  020412            4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
032600  001403                    BEQ     TST463          ;;BR IF YES
                        
032602  005003                    CLR     R3              ;GET THE WAS DATA
032604  051203                    BIS     (R2),R3
032606  104001            5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
                        
                        ;;*****************************************************************
                        ;*TEST 463       MOV SM2,DM7 TEST - N:C = 0100
                        ;;*****************************************************************
032610                  TST463:
032610  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
032612  012700  000463            MOV     #463,R0         ;;LOAD R0 WITH TEST NUMBER
032616  013701  032650            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
032622  005004                    CLR     R4              ;RESULT S / B = 177777
032624  005104                    COM     R4
032626  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
032632  012705  063324            MOV     #DWTA+2,R5      ;SOURCE ADDR = DWTA+2
032636  012703  063276            MOV     #ATA,R3         ;BASE DEST ADDR   ATA
032642  005012                    CLR     (R2)            ;MAKE [DEST] = 000000
032644  000257                    CCC                     ;CLEAR FLAGS
032646  000264                    264                     ;N:C - 0100
                        
032650  011573  000010    2$:     MOV     (R5),@10(R3)    ;TEST THE MOV - SM2,DM7
                        
032654  100003                    BPL     3$              ;N:C   '000 ?
032656  001402                    BEQ     3$
032660  102401                    BVS     3$
032662  103001                    BCC     4$
```

```
 9427  032664  104001                  3$:     ERROR   1               ;MOV FAILED TO ALTER CODES PROPERLY
 9428
 9429  032666  020412                  4$:     CMP     R4,(R2)         ;RESULT CORRECT ??
 9430  032670  001403                          BEQ     TST464          ;;BR IF YES
 9431
 9432  032672  005003                          CLR     R3              ;GET THE WAS DATA
 9433  032674  051203                          BIS     (R2),R3
 9434  032676  104001                  5$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9435
 9436                                  ;;*********************************************************************
 9437                                  ;*TEST 464       MOV  SM0,DM1 TEST
 9438                                  ;;*********************************************************************
 9439  032700                          TST464:
 9440  032700  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9441  032702  012700  000464                  MOV     #464,R0         ;;LOAD R0 WITH TEST NUMBER
 9442  032706  013701  032726                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9443  032712  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9444  032716  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9445  032720  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
 9446  032722  005012                          CLR     (R2)            ;[DEST] = 000000
 9447  032724  000257                          CCC                     ;SCOPE SYNC
 9448
 9449  032726  010015                  2$:     MOV     R0,(R5)         ;TEST THE MOV
 9450
 9451  032730  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9452  032732  001402                          BEQ     TST465          ;;BR IF YES
 9453
 9454  032734  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9455  032736  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9456
 9457                                  ;;*********************************************************************
 9458                                  ;*TEST 465       MOV  SM0,DM2 TEST
 9459                                  ;;*********************************************************************
 9460  032740                          TST465:
 9461  032740  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9462  032742  012700  000465                  MOV     #465,R0         ;;LOAD R0 WITH TEST NUMBER
 9463  032746  013701  032766                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9464  032752  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9465  032756  010004                          MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9466  032760  010205                          MOV     R2,R5           ;R5 GETS DEST ADDR
 9467  032762  005012                          CLR     (R2)            ;[DEST] = 000000
 9468  032764  000257                          CCC                     ;SCOPE SYNC
 9469
 9470  032766  010025                  2$:     MOV     R0,(R5)+        ;TEST THE MOV
 9471
 9472  032770  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9473  032772  001402                          BEQ     TST466          ;;BR IF YES
 9474
 9475  032774  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9476  032776  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9477
 9478                                  ;;*********************************************************************
 9479                                  ;*TEST 466       MOV  SM0,DM3 TEST
 9480                                  ;;*********************************************************************
 9481  033000                          TST466:
 9482  033000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9483  033002  012700  000466          MOV     #466,R0         ;;LOAD R0 WITH TEST NUMBER
 9484  033006  013701  033030          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9485  033012  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9486  033016  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9487  033020  012705  063306          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9488  033024  005012                  CLR     (R2)            ;[DEST] = 000000
 9489  033026  000257                  CCC                     ;SCOPE SYNC
 9490
 9491  033030  010035          2$:     MOV     R0,@(R5)+       ;TEST THE MOV
 9492
 9493  033032  020412                  CMP     R4,(R2)         ;CORRECT RESULT
 9494  033034  001402                  BEQ     TST467          ;;BR IF YES
 9495
 9496  033036  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9497  033040  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9498
 9499                          ;;*****************************************************************
 9500                          ;*TEST 467       MOV  SM0,DM4 TEST
 9501                          ;;*****************************************************************
 9502  033042                  TST467:
 9503  033042  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9504  033044  012700  000467          MOV     #467,R0         ;;LOAD R0 WITH TEST NUMBER
 9505  033050  013701  033072          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9506  033054  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9507  033060  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9508  033062  012705  063314          MOV     #MBUF0+2,R5     ;R5 CONTAINS BASE DEST ADDR
 9509  033066  005012                  CLR     (R2)            ;[DEST] = 000000
 9510  033070  000257                  CCC                     ;SCOPE SYNC
 9511
 9512  033072  010045          2$:     MOV     R0,-(R5)        ;TEST THE MOV
 9513
 9514  033074  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9515  033076  001402                  BEQ     TST470          ;;BR IF YES
 9516
 9517  033100  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9518  033102  104001          3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9519
 9520                          ;;*****************************************************************
 9521                          ;*TEST 470       MOV  SM0,DM5 TEST
 9522                          ;;*****************************************************************
 9523  033104                  TST470:
 9524  033104  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9525  033106  012700  000470          MOV     #470,R0         ;;LOAD R0 WITH TEST NUMBER
 9526  033112  013701  033134          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9527  033116  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9528  033122  010004                  MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9529  033124  012705  063310          MOV     #ATA+12,R5      ;R5 CONTAINS BASE DEST ADDR
 9530  033130  005012                  CLR     (R2)            ;[DEST] = 000000
 9531  033132  000257                  CCC                     ;SCOPE SYNC
 9532
 9533  033134  010055          2$:     MOV     R0,@-(R5)       ;TEST THE MOV
 9534
 9535  033136  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9536  033140  001402                  BEQ     TST471          ;;BR IF YES
 9537
 9538  033142  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
 9539  033144  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9540
 9541
 9542                                  ;;*****************************************************************
 9543                                  ;*TEST 471      MOV  SM0,DM6 TEST
 9544                                  ;;*****************************************************************
 9545  033146                          TST471:
 9546  033146  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9547  033150  012700  000471                 MOV     #471,R0         ;;LOAD R0 WITH TEST NUMBER
 9548  033154  013701  033176                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9549  033160  012702  063316                 MOV     #MBUF1,R2       ;DEST ADDR = MBUF1
 9550  033164  010004                         MOV     R0,R4           ;RESULT S / B = TEST NUMBER
 9551  033166  012705  063312                 MOV     #MBUF0,R5       ;BASE DEST ADDR = MBUF0
 9552  033172  005012                         CLR     (R2)            ;[DEST] = 000000
 9553  033174  000257                         CCC                     ;SCOPE SYNC
 9554
 9555  033176  010065  000004         2$:     MOV     R0,4(R5)        ;TEST THE MOV
 9556
 9557  033202  020412                         CMP     R4,(R2)         ;RESULT CORRECT ?
 9558  033204  001402                         BEQ     TST472          ;;BR IF YES
 9559
 9560  033206  011203                         MOV     (R2),R3         ;GET THE WAS DATA
 9561  033210  104001                  3$:     ERROR   1               ;MOV DELIVERED THE WRONG RESULT
 9562
 9563                                  ;;*****************************************************************
 9564                                  ;*TEST 472      MOVB TEST - SM0,DM0 - EXTEND 1'S
 9565                                  ;;*****************************************************************
 9566  033212                          TST472:
 9567  033212  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9568  033214  012700  000472                 MOV     #472,R0         ;;LOAD R0 WITH TEST NUMBER
 9569  033220  013701  033242                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9570  033224  012704  177652                 MOV     #177652,R4      ;RESULT S / B = 177652
 9571  033230  012705  000252                 MOV     #252,R5         ;SOURCE OP = 252
 9572  033234  005003                         CLR     R3              ;[DEST] = 000000
 9573  033236  000257                         CCC                     ;CLEAR FLAGS
 9574  033240  000266                         266                     ;N:C = 0110
 9575
 9576  033242  110503                  2$:     MOVB    R5,R3           ;TEST THE MOVB
 9577
 9578  033244  100003                         BPL     3$              ;N:C = 1000 ?
 9579  033246  001402                         BEQ     3$
 9580  033250  102401                         BVS     3$
 9581  033252  103001                         BCC     4$
 9582
 9583  033254  104002                  3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9584
 9585  033256  020403                  4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9586  033260  001401                         BEQ     TST473          ;;BR IF YES
 9587
 9588  033262  104002                  5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9589
 9590                                  ;;*****************************************************************
 9591                                  ;*TEST 473      MOVB TEST - SM0,DM0 - EXTEND 0'S
 9592                                  ;;*****************************************************************
 9593  033264                          TST473:
 9594  033264  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
 9595  033266  012700  000473          MOV     #473,R0         ;;LOAD R0 WITH TEST NUMBER
 9596  033272  013701  033314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9597  033276  005004                  CLR     R4              ;RESULT S / B = 000000
 9598  033300  012705  177400          MOV     #177400,R5      ;SOURCE OP = 177400
 9599  033304  005C03                  CLR     R3              ;[DEST] = 177777
 9600  033306  005103                  COM     R3
 9601  033310  000257                  CCC                     ;CLEAR FLAGS
 9602  033312  000271                  271                     ;N:C = 1001
 9603
 9604  033314  110503          2$:     MOVB    R5,R3           ;TEST THE MOVB
 9605
 9606  033316  100403                  BMI     3$              ;N:C = 0101 ?
 9607  033320  001002                  BNE     3$
 9608  033322  102401                  BVS     3$
 9609  033324  103401                  BCS     4$
 9610
 9611  033326  104002          3$:     ERROR   2               ;MOVB FAILED TO ALTER CODES PROPERLY
 9612
 9613  033330  020403          4$:     CMP     R4,R3           ;RESULT CORRECT ?
 9614  033332  001401                  BEQ     TST474          ;;BR IF YES
 9615
 9616  033334  104002          5$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9617
 9618                          ;;*********************************************************
 9619                          ;*TEST 474       MOVB TEST - SM1,DM0 - SOURCE ADDR EVEN
 9620                          ;;*********************************************************
 9621  033336          TST474:
 9622  033336  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9623  033340  012700  000474          MOV     #474,R0         ;;LOAD R0 WITH TEST NUMBER
 9624  033344  013701  033364          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9625  033350  005004                  CLR     R4              ;RESULT S / B = 000000
 9626  033352  012705  064630          MOV     #DBTA,R5        ;SOURCE ADDR - DBTA
 9627  033356  005003                  CLR     R3              ;[DEST] = 177777
 9628  033360  005103                  COM     R3
 9629  033362  000257                  CCC                     ;SCOPE SYNC
 9630
 9631  033364  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
 9632
 9633  033366  020403                  CMP     R4,R3           ;RESULT CORRECT ?
 9634  033370  001401                  BEQ     TST475          ;;BR IF YES
 9635
 9636  033372  104002          3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9637
 9638                          ;;*********************************************************
 9639                          ;*TEST 475       MOVB TEST - SM1,DM0 - SOURCE ADDR ODD
 9640                          ;;*********************************************************
 9641  033374          TST475:
 9642  033374  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9643  033376  012700  000475          MOV     #475,R0         ;;LOAD R0 WITH TEST NUMBER
 9644  033402  013701  033424          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9645  033406  012704  000125          MOV     #125,R4         ;RESULT S / B = 125
 9646  033412  012705  064633          MOV     #DBTA+3,R5      ;SOURCE ADDR = DBTA+3
 9647  033416  012703  177400          MOV     #177400,R3      ;[DEST] = 177400
 9648  033422  000257                  CCC                     ;SCOPE SYNC
 9649
 9650  033424  111503          2$:     MOVB    (R5),R3         ;TEST THE MOVB
```

```
 9651
 9652  033426  020403                    CMP     R4,R3           ;RESULT CORRECT ?
 9653  033430  001401                    BEQ     TST476          ;;BR IF YES
 9654
 9655  033432  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9656
 9657                            ;;*************************************************************
 9658                            ;*TEST 476       MOVB TEST - SM2,DM0 - SOURCE ADDR ODD
 9659                            ;;*************************************************************
 9660  033434                    TST476:
 9661  033434  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9662  033436  012700  000476            MOV     #476,R0         ;;LOAD R0 WITH TEST NUMBER
 9663  033442  013701  033462            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9664  033446  012704  177777            MOV     #-1,R4          ;RESULT S / B = 177777
 9665  033452  012705  064631            MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 9666  033456  005003                    CLR     R3              ;[DEST] = 000000
 9667  033460  000257                    CCC                     ;SCOPE SYNC
 9668
 9669  033462  112503            2$:     MOVB    (R5)+,R3        ;TEST THE MOVB
 9670
 9671  033464  020403                    CMP     R4,R3           ;RESULT CORRECT ?
 9672  033466  001401                    BEQ     4$              ;BR IF YES
 9673
 9674  033470  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9675
 9676  033472  022705  064632    4$:     CMP     #DBTA+2,R5      ;DID MOVB INCREMENT SRC REG ?
 9677  033476  001401                    BEQ     TST477          ;;BR IF YES
 9678
 9679  033500  104005            5$:     ERROR   5               ;MOVB FAILED TO UPDATE SRC REG
 9680
 9681                            ;;*************************************************************
 9682                            ;*TEST 477       MOVB TEST - SM2,DM0 - SOURCE ADDR EVEN
 9683                            ;;*************************************************************
 9684  033502                    TST477:
 9685  033502  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9686  033504  012700  000477            MOV     #477,R0         ;;LOAD R0 WITH TEST NUMBER
 9687  033510  013701  033530            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9688  033514  005004                    CLR     R4              ;RESULT S / B = 000000
 9689  033516  012705  064630            MOV     #DBTA,R5        ;SOURCE ADDR = DBTA
 9690  033522  012703  177400            MOV     #177400,R3      ;[DEST] = 177400
 9691  033526  000257                    CCC                     ;SCOPE SYNC
 9692
 9693  033530  112503            2$:     MOVB    (R5)+,R3                ;TEST THE MOVB
 9694
 9695  033532  020403                    CMP     R4,R3           ;RESULT CORRECT ?
 9696  033534  001401                    BEQ     4$              ;BR IF YES
 9697
 9698  033536  104002            3$:     ERROR   2               ;MOVB DELIVERED THE WRONG RESULT
 9699
 9700  033540  022705  064631    4$:     CMP     #DBTA+1,R5      ;DID MOVB INCREMENT SRC REG ?
 9701  033544  001401                    BEQ     TST500          ;;BR IF YES
 9702
 9703  033546  104005            5$:     ERROR   5               ;MOVB FAILED TO UPDATE SOURCE REG
 9704
 9705                            ;;*************************************************************
 9706                            ;*TEST 500       MOVB TEST - SM1,DM1 - SRC ADR ODD / DST ADR EVEN
```

```
 9707
 9708  033550
 9709  033550  000004                  ;;*****************************************************************
 9710  033552  012700  000500  TST500:
 9711  033556  013701  033602          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9712  033562  012702  063312          MOV     #500,R0         ;;LOAD R0 WITH TEST NUMBER
 9713  033566  012704  000377          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9714  033572  012705  064631          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9715  033576  005012                  MOV     #377,R4         ;RESULT S / B = 377
 9716  033600  000257                  MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9717                                  CLR     (R2)            ;[DEST] = 000000
 9718  033602  111512          2$:     CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9719                                  MOVB    (R5),(R2)       ;TEST THE MOVB
 9720  033604  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9721  033606  001402                  BEQ     TST501          ;;BR IF YES
 9722
 9723  033610  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9724  033612  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9725
 9726                                  ;;*****************************************************************
 9727                                  ;*TEST 501      MOVB TEST - SM1,DM2 - SRC ADR ODD / DST ADR EVEN
 9728                                  ;;*****************************************************************
 9729  033614                  TST501:
 9730  033614  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9731  033616  012700  000501          MOV     #501,R0         ;;LOAD R0 WITH TEST NUMBER
 9732  033622  013701  033650          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9733  033626  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9734  033632  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9735  033636  012705  064631          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9736  033642  005012                  CLR     (R2)            ;[DEST] = 000000
 9737  033644  010203                  MOV     R2,R3           ;[R3] = DEST ADDR
 9738  033646  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9739
 9740  033650  111523          2$:     MOVB    (R5),(R3)+      ;TEST THE MOVB
 9741
 9742  033652  020412                  CMP     R4,(R2)         ;CORRECT RESULT ?
 9743  033654  001402                  BEQ     4$              ;BR IF YES
 9744
 9745  033656  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9746  033660  104001          3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9747
 9748  033662  022703  063313  4$:     CMP     #MBUF0+1,R3     ;DID MOVB INCREMENT THE DEST REG ?
 9749  033666  001401                  BEQ     TST502          ;;BR IF YES
 9750
 9751  033670  104005          5$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9752
 9753                                  ;;*****************************************************************
 9754                                  ;*TEST 502      MOVB TEST - SM1,DM3 - SRC ADR ODD / DST ADR EVEN
 9755                                  ;;*****************************************************************
 9756  033672                  TST502:
 9757  033672  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9758  033674  012700  000502          MOV     #502,R0         ;;LOAD R0 WITH TEST NUMBER
 9759  033700  013701  033730          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9760  033704  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9761  033710  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9762  033714  012705  064631          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
```

```
 9763  033720  005012                  CLR     (R2)            ;[DEST] = 000000
 9764  033722  012703  063306          MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA +10
 9765  033726  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9766
 9767  033730  111533          2$:     MOVB    (R5),@(R3)+     ;TEST THE MOVB
 9768
 9769  033732  022703  063310          CMP     #ATA+12,R3      ;DID DEST REG GET INCREMENTED ?
 9770  033736  001401                  BEQ     4$              ;BR IF YES
 9771
 9772  033740  104005          3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9773
 9774  033742  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9775  033744  001402                  BEQ     TST503          ;;BR IF YES
 9776
 9777  033746  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9778  033750  104001          5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9779
 9780                          ;;*****************************************************************
 9781                          ;*TEST 503      MOVB TEST - SM1,DM4 - SRC ADR ODD / DST ADR EVEN
 9782                          ;;*****************************************************************
 9783  033752                  TST503:
 9784  033752  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9785  033754  012700  000503          MOV     #503,R0         ;;LOAD R0 WITH TEST NUMBER
 9786  033760  013701  034010          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9787  033764  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9788  033770  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9789  033774  012705  064631          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9790  034000  005012                  CLR     (R2)            ;[DEST] = 000000
 9791  034002  012703  063313          MOV     #MBUF0+1,R3     ;INITIAL DEST ADDR = MBUF0+1
 9792  034006  000257                  CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9793
 9794  034010  111543          2$:     MOVB    (R5),-(R3)      ;TEST THE MOVB
 9795
 9796  034012  020302                  CMP     R3,R2           ;DID MOVB DECREMENT DEST REG ?
 9797  034014  001401                  BEQ     4$              ;BR IF YES
 9798
 9799  034016  104005          3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9800
 9801  034020  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9802  034022  001402                  BEQ     TST504          ;;BR IF YES
 9803
 9804  034024  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9805  034026  104001          5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9806
 9807                          ;;*****************************************************************
 9808                          ;*TEST 504      MOVB TEST - SM1,DM5 - SRC ADR ODD / DST ADR EVEN
 9809                          ;;*****************************************************************
 9810  034030                  TST504:
 9811  034030  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9812  034032  012700  000504          MOV     #504,R0         ;;LOAD R0 WITH TEST NUMBER
 9813  034036  013701  034066          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9814  034042  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9815  034046  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9816  034052  012705  064631          MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9817  034056  005012                  CLR     (R2)            ;[DEST] = 000000
 9818  034060  012703  063310          MOV     #ATA+12,R3      ;INITIAL DEST ADDR = ATA +12
```

```
 9819  034064  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9820
 9821  034066  111553            2$:     MOVB    (R5),@-(R3)     ;TEST THE MOVB
 9822
 9823  034070  022703  063306            CMP     #ATA+10,R3      ;DID MOVB DECREMENT DEST REG ?
 9824  034074  001401                    BEQ     4$              ;BR IF YES
 9825
 9826  034076  104005            3$:     ERROR   5               ;MOVB FAILED TO UPDATE DEST REG
 9827
 9828  034100  020412            4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 9829  034102  001402                    BEQ     TST505          ;;BR IF YES
 9830
 9831  034104  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9832  034106  104001            5$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9833
 9834                            ;;*********************************************************
 9835                            ;*TEST 505      MOVB TEST - SM1,DM6 - SRC ADR ODD / DST ADR EVEN
 9836                            ;;*********************************************************
 9837  034110                    TST505:
 9838  034110  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9839  034112  012700  000505            MOV     #505,R0         ;;LOAD R0 WITH TEST NUMBER
 9840  034116  013701  034146            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9841  034122  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9842  034126  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9843  034132  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9844  034136  005012                    CLR     (R2)            ;[DEST] = 000000
 9845  034140  012703  063320            MOV     #MBUF0+6,R3     ;BASE DEST ADDR = MBUF0+6
 9846  034144  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9847
 9848  034146  111563  177772    2$:     MOVB    (R5),-6(R3)     ;TEST THE MOVB
 9849
 9850  034152  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9851  034154  001402                    BEQ     TST506          ;;BR IF YES
 9852
 9853  034156  011203                    MOV     (R2),R3         ;GET THE WAS DATA
 9854  034160  104001            3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9855
 9856                            ;;*********************************************************
 9857                            ;*TEST 506      MOVB TEST - SM1,DM7 - SRC ADR ODD / DST ADR EVEN
 9858                            ;;*********************************************************
 9859  034162                    TST506:
 9860  034162  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9861  034164  012700  000506            MOV     #506,R0         ;;LOAD R0 WITH TEST NUMBER
 9862  034170  013701  034220            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9863  034174  012702  063312            MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9864  034200  012704  000377            MOV     #377,R4         ;RESULT S / B = 377
 9865  034204  012705  064631            MOV     #DBTA+1,R5      ;SRC ADDR = DBTA +1
 9866  034210  005012                    CLR     (R2)            ;[DEST] = 000000
 9867  034212  012703  063276            MOV     #ATA,R3         ;BASE DEST ADDR = ATA
 9868  034216  000257                    CCC                     ;CLEAR FLAGS - SCOPE SYNC
 9869
 9870  034220  111573  000010    2$.     MOVB    (R5),@10(R3)    ;TEST THE MOVB
 9871
 9872  034224  020412                    CMP     R4,(R2)         ;CORRECT RESULT ?
 9873  034226  001402                    BEQ     TST507          ;;BR IF YES
 9874
```

```
 9875  034230  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9876  034232  104001                  3$:     ERROR   1               ;MOVB DELIVERED WRONG RESULT
 9877
 9878                                  ;;*****************************************************************
 9879                                  ;*TEST 507       MOVB  SM0,DM1 TEST
 9880                                  ;;*****************************************************************
 9881  034234                          TST507:
 9882  034234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9883  034236  012700  000507                  MOV     #507,R0         ;;LOAD R0 WITH TEST NUMBER
 9884  034242  013701  034270                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9885  034246  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9886  034252  012704  000377                  MOV     #377,R4         ;RESUL' S / B = 377
 9887  034256  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9888  034262  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9889  034264  005012                          CLR     (R2)            ;[DEST] = 000000
 9890  034266  000257                          CCC                     ;SCOPE SYNC
 9891
 9892  034270  110315                  2$:     MOVB    R3,(R5)         ;TEST THE MOVB
 9893
 9894  034272  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9895  034274  001402                          BEQ     TST510          ;;BR IF YES
 9896
 9897  034276  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9898  034300  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9899
 9900                                  ;;*****************************************************************
 9901                                  ;*TEST 510       MOVB  SM0,DM2 TEST
 9902                                  ;;*****************************************************************
 9903  034302                          TST510:
 9904  034302  000004                          SCOPE                   ;CA L THE SCOPE LOOP UTILITY
 9905  034304  012700  000510                  MOV     #510,R0         ;;LOAD R0 WITH TEST NUMBER
 9906  034310  013701  034336                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9907  034314  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9908  034320  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 9909  034324  012703  177777                  MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9910  034330  010205                          MOV     R2,R5           ;R5 CONTAINS DEST ADDR
 9911  034332  005012                          CLR     (R2)            ;[DEST] = 000000
 9912  034334  000257                          CCC                     ;SCOPE SYNC
 9913
 9914  034336  110325                  2$:     MOVB    R3,(R5)+        ;TEST THE MOVB
 9915
 9916  034340  020412                          CMP     R4,(R2)         ;RESULT CORRECT ?
 9917  034342  001402                          BEQ     TST511          ;;BR IF YES
 9918
 9919  034344  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 9920  034346  104001                  3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9921
 9922                                  ;;*****************************************************************
 9923                                  ;*TEST 511       MOVB  SM0,DM3 TEST
 9924                                  ;;*****************************************************************
 9925  034350                          TST511:
 9926  034350  000C04                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9927  034352  012700  000511                  MOV     #511,R0         ;;LOAD R0 WITH TEST NUMBER
 9928  034356  013701  034406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9929  034362  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9930  034366  012704  000377                  MOV     #377,R4         ;RESULT S / B - 377
```

```
 9931  034372  012703  177777          MOV     #-1,R3          ;SOURCE OP IN R3
 9932  034376  012705  063306          MOV     #ATA+10,R5      ;BASE DEST ADDR = ATA+10
 9933  034402  005012                  CLR     (R2)            ;[DEST] = 000000
 9934  034404  000257                  CCC                     ;SCOPE SYNC
 9935
 9936  034406  110335          2$:     MOVB    R3,@(R5)+       ;TEST THE MOVB
 9937
 9938  034410  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9939  034412  001402                  BEQ     TST512          ;;BR IF YES
 9940
 9941  034414  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9942  034416  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9943
 9944                          ;;*****************************************************************
 9945                          ;*TEST 512       MOVB  SM0,DM4 TEST
 9946                          ;;*****************************************************************
 9947  034420                  TST512:
 9948  034420  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9949  034422  012700  000512          MOV     #512,R0         ;;LOAD R0 WITH TEST NUMBER
 9950  034426  013701  034456          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9951  034432  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9952  034436  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
 9953  034442  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9954  034446  012705  063314          MOV     #MBUF0+2,R5     ;BASE DEST ADDR = MBUF0+2
 9955  034452  005012                  CLR     (R2)            ;[DEST] = 000000
 9956  034454  000257                  CCC                     ;SCOPE SYNC
 9957
 9958  034456  110345          2$:     MOVB    R3,-(R5)        ;TEST THE MOVB
 9959
 9960  034460  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9961  034462  001402                  BEQ     TST513          ;;BR IF YES
 9962
 9963  034464  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9964  034466  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
 9965
 9966                          ;;*****************************************************************
 9967                          ;*TEST 513       MOVB  SM0,DM6 TEST
 9968                          ;;*****************************************************************
 9969  034470                  TST513:
 9970  034470  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9971  034472  012700  000513          MOV     #513,R0         ;;LOAD R0 WITH TEST NUMBER
 9972  034476  013701  034526          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9973  034502  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 9974  034506  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
 9975  034512  012703  177777          MOV     #-1,R3          ;R3 CONTAINS SOURCE OP
 9976  034516  012705  063314          MOV     #MBUF0+2,R5     ;BASE DEST ADDR - MBUF0+2
 9977  034522  005012                  CLR     (R2)            ;[DEST] = 000000
 9978  034524  000257                  CCC                     ;SCOPE SYNC
 9979
 9980  034526  110365  177776  2$:     MOVB    R3,-2(R5)       ;TEST THE MOVB
 9981
 9982  034532  020412                  CMP     R4,(R2)         ;RESULT CORRECT ?
 9983  034534  001402                  BEQ     TST514          ;;BR IF YES
 9984
 9985  034536  011203                  MOV     (R2),R3         ;GET THE WAS DATA
 9986  034540  104001          3$:     ERROR   1               ;MOVB DELIVERED THE WRONG RESULT
```

```
 9987
 9988                                   ;;*****************************************************************
 9989                                   ;*TEST 514      BIS TEST - SM0,DM0 - N:C = 0111
 9990                                   ;;*****************************************************************
 9991  034542                           TST514:
 9992  034542  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 9993  034544  012700  000514                  MOV     #514,R0         ;;LOAD R0 WITH TEST NUMBER
 9994  034550  013701  034574                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 9995  034554  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
 9996  034560  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
 9997  034564  012703  052525                  MOV     #52525,R3       ;[DEST] = 52525
 9998  034570  000257                          CCC                     ;CLEAR FLAGS
 9999  034572  000267                          267                     ;N:C = 0111
10000
10001  034574  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10002
10003  034576  100003                          BPL     3$              ;N:C = 1001 ?
10004  034600  001402                          BEQ     3$
10005  034602  102401                          BVS     3$
10006  034604  103401                          BCS     4$
10007
10008  034606  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10009
10010  034610  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10011  034612  001401                          BEQ     TST515          ;;BR IF YES
10012
10013  034614  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10014
10015                                   ;;*****************************************************************
10016                                   ;*TEST 515      BIS TEST - SM0,DM0 - N:C = 1000
10017                                   ;;*****************************************************************
10018  034616                           TST515:
10019  034616  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10020  034620  012700  000515                  MOV     #515,R0         ;;LOAD R0 WITH TEST NUMBER
10021  034624  013701  034654                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10022                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT10
10023  034630  032737  002000  063234          BIT     #BIT10,@#BPTLOC ;BREAKPOINT HALT SET ??
10024  034636  001401                          BEQ     .+4             ;BR IF NOT
10025  034640  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10026  034642  005004                          CLR     R4              ;RESULT S / B = 000000
10027  034644  005005                          CLR     R5              ;SRC OPR = 000000
10028  034646  005003                          CLR     R3              ;[DEST] = 000000
10029  034650  000257                          CCC                     ;CLEAR FLAGS
10030  034652  000270                          SEN                     ;N:C = 1000
10031
10032  034654  050503                  2$:     BIS     R5,R3           ;TEST THE BIS
10033
10034  034656  100403                          BMI     3$              ;N:C = 0100
10035  034660  001002                          BNE     3$
10036  034662  102401                          BVS     3$
10037  034664  103001                          BCC     4$
10038
10039  034666  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10040
10041  034670  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10042  034672  001401                          BEQ     TST516          ;;BR IF YES
```

```
10043
10044  034674  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10045
10046                                  ;;*****************************************************************
10047                                  ;*TEST 516      BIC TEST - SM0,DM0 - N:C = 0111
10048                                  ;;*****************************************************************
10049  034676                          TST516:
10050  034676  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10051  034700  012700  000516                 MOV     #516,R0         ;;LOAD R0 WITH TEST NUMBER
10052  034704  013701  034730                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10053  034710  012704  100000                 MOV     #100000,R4      ;RESULT S / B = 100000
10054  034714  012705  077777                 MOV     #77777,R5       ;SRC OPR = 77777
10055  034720  012703  177777                 MOV     #-1,R3          ;[DEST] = 177777
10056  034724  000257                         CCC                     ;CLEAR FLAGS
10057  034726  000267                         267                     ;N:C = 0111
10058
10059  034730  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10060
10061  034732  100003                         BPL     3$              ;N:C = 1001 ?
10062  034734  001402                         BEQ     3$
10063  034736  102401                         BVS     3$
10064  034740  103401                         BCS     4$
10065
10066  034742  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10067
10068  034744  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10069  034746  001401                         BEQ     TST517          ;;BR IF YES
10070
10071  034750  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10072
10073                                  ;;*****************************************************************
10074                                  ;*TEST 517      BIC TEST - SM0,DM0 - N:C = 1000
10075                                  ;;*****************************************************************
10076  034752                          TST517:
10077  034752  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10078  034754  012700  000517                 MOV     #517,R0         ;;LOAD R0 WITH TEST NUMBER
10079  034760  013701  034776                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10080  034764  005004                         CLR     R4              ;RESULT S / B = 000000
10081  034766  005005                         CLR     R5              ;SRC OPR = 000000
10082  034770  005003                         CLR     R3              ;[DEST] = 000000
10083  034772  000257                         CCC                     ;CLEAR FLAGS
10084  034774  000270                         SEN                     ;N:C = 1000
10085
10086  034776  040503                  2$:     BIC     R5,R3           ;TEST THE BIC
10087
10088  035000  100403                         BMI     3$              ;N:C = 0100
10089  035002  001002                         BNE     3$
10090  035004  102401                         BVS     3$
10091  035006  103001                         BCC     4$
```

```
10092
10093  035010  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10094
10095  035012  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10096  035014  001401                          BEQ     TST520          ;;BR IF YES
10097
10098  035016  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10099
10100                                  ;;*********************************************************
10101                                  ;*TEST 520       BIT TEST - SM0,DM0 - N:C = 0111
10102                                  ;;*********************************************************
10103  035020                          TST520:
10104  035020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10105  035022  012700  000520                  MOV     #520,R0         ;;LOAD R0 WITH TEST NUMBER
10106  035026  013701  035052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10107  035032  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10108  035036  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10109  035042  012703  100000                  MOV     #100000,R3      ;[DEST] - 100000
10110  035046  000257                          CCC                     ;CLEAR FLAGS
10111  035050  000267                          267                     ;N:C = 0111
10112
10113  035052  030503                  2$:     BIT     R5,R3           ;TEST THE BIT
10114
10115  035054  100003                          BPL     3$              ;N:C = 1001
10116  035056  001402                          BEQ     3$
10117  035060  102401                          BVS     3$
10118  035062  103401                          BCS     4$
10119
10120  035064  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10121
10122  035066  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10123  035070  001402                          BEQ     TST521          ;;BR IF YES
10124
10125  035072  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10126  035074  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10127
10128                                  ;;*********************************************************
10129                                  ;*TEST 521       BIT TEST - SM0,DM0 - N:C = 1000
10130                                  ;;*********************************************************
10131  035076                          TST521:
10132  035076  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10133  035100  012700  000521                  MOV     #521,R0         ;;LOAD R0 WITH TEST NUMBER
10134  035104  013701  035126                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10135  035110  012704  125252                  MOV     #125252,R4      ;RESULT S / B = 125252
10136  035114  012705  052525                  MOV     #52525,R5       ;SRC OPR = 52525
10137  035120  010403                          MOV     R4,R3           ;[DEST] = 125252
10138  035122  000257                          CCC                     ;CLEAR FLAGS
10139  035124  000270                          SEN                     ;N:C = 1000
10140
10141  035126  030503                  2$:     BIT     R5,R3   ;TEST THE BIT
10142
10143  035130  100403                          BMI     3$              ;N:C = 0100
10144  035132  001002                          BNE     3$
10145  035134  102401                          BVS     3$
10146  035136  103001                          BCC     4$
10147
```

```
10148  035140  104002                   3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10149
10150  035142  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10151  035144  001401                           BEQ     TST522          ;;BR IF YES
10152
10153  035146  104002                   5$:     ERROR   2               ;BIT DELIVERED A RESULT
10154
10155                                   ;;*********************************************************************
10156                                   ;*TEST 522       CMP TEST - SM0,DM0 - N:C = 0110
10157                                   ;;*********************************************************************
10158  035150                           TST522:
10159  035150  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10160  035152  012700  000522                   MOV     #522,R0         ;;LOAD R0 WITH TEST NUMBER
10161  035156  013701  035200                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10162  035162  012704  000001                   MOV     #+1,R4          ;RESULT S / B = +1
10163  035166  005005                           CLR     R5              ;SRC OPR = 000000
10164  035170  012703  000001                   MOV     #+1,R3          ;[DEST] = +1
10165  035174  000257                           CCC                     ;CLEAR FLAGS
10166  035176  000266                           266                     ;N:C = 0110
10167
10168  035200  020503                   2$:     CMP     R5,R3           ;TEST THE CMP
10169
10170  035202  100003                           BPL     3$              ;N:C = 1001
10171  035204  001402                           BEQ     3$
10172  035206  102401                           BVS     3$
10173  035210  103401                           BCS     4$
10174
10175  035212  104002                   3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10176
10177  035214  020403                   4$:     CMP     R4,R3           ;CORRECT RESULT ?
10178  035216  001401                           BEQ     TST523          ;;BR IF YES
10179
10180  035220  104002                   5$:     ERROR   2               ;CMP DELIVERED A RESULT
10181
10182                                   ;;*********************************************************************
10183                                   ;*TEST 523       CMP TEST - SM0,DM0 - N:C = 1010
10184                                   ;;*********************************************************************
10185  035222                           TST523:
10186  035222  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10187  035224  012700  000523                   MOV     #523,R0         ;;LOAD R0 WITH TEST NUMBER
10188  035230  013701  035252                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10189  035234  012704  177777                   MOV     #-1,R4          ;RESULT S / B = 177777
10190  035240  012705  177777                   MOV     #-1,R5          ;SRC OPR = 177777
10191  035244  010403                           MOV     R4,R3           ;[DEST]  177777
10192  035246  000257                           CCC                     ;CLEAR FLAGS
10193  035250  000272                           272                     ;N:C = 1010
10194
10195  035252  020503                   2$:     CMP     R5,R3           ;TEST THE CMP
10196
10197  035254  100403                           BMI     3$              ;N:C = 0100
10198  035256  001002                           BNE     3$
10199  035260  102401                           BVS     3$
10200  035262  103001                           BCC     4$
10201
10202  035264  104002                   3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10203
```

```
10204  035266  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10205  035270  001401                          BEQ     TST524          ;;BR IF YES
10206
10207  035272  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10208
10209                                  ;;*********************************************************
10210                                  ;*TEST 524       CMP TEST - SM0,DM0 - N:C = 0000
10211                                  ;;*********************************************************
10212  035274                          TST524:
10213  035274  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10214  035276  012700  000524                  MOV     #524,R0         ;;LOAD R0 WITH TEST NUMBER
10215  035302  013701  035324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10216  035306  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10217  035312  012705  100000                  MOV     #100000,R5                      ;SRC OPR   100000
10218  035316  012703  000001                  MOV     #+1,R3          ;[DEST] - +1
10219  035322  000257                          CCC                     ;CLEAR FLAGS
10220
10221  035324  020503                  2$:     CMP     R5,R3           ;TEST THE CMP
10222
10223  035326  100403                          BMI     3$              ;N:C = 0010
10224  035330  001402                          BEQ     3$
10225  035332  102001                          BVC     3$
10226  035334  103001                          BCC     4$
10227
10228  035336  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10229
10230  035340  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10231  035342  001401                          BEQ     TST525          ;;BR IF YES
10232
10233  035344  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10234
10235                                  ;;*********************************************************
10236                                  ;*TEST 525       BIS TEST - SM0,DM1 - N:C = 0111
10237                                  ;;*********************************************************
10238  035346                          TST525:
10239  035346  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10240  035350  012700  000525                  MOV     #525,R0         ;;LOAD R0 WITH TEST NUMBER
10241  035354  013701  035404                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10242  035360  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10243  035364  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10244  035370  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
10245  035374  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
10246  035400  000257                          CCC                     ;CLEAR FLAGS
10247  035402  000267                          267                     ;N:C = 0111
10248
10249  035404  050512                  2$:     BIS     R5,(R2)         ;TEST THE BIS
10250
10251  035406  100003                          BPL     3$              ;N:C = 1001
10252  035410  001402                          BEQ     3$
10253  035412  102401                          BVS     3$
10254  035414  103401                          BCS     4$
10255
10256  035416  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10257
10258  035420  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10259  035422  001402                          BEQ     TST526          ;;BR IF YES
```

```
10260
10261  035424  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10262  035426  104001          5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10263
10264                          ;;*****************************************************************
10265                          ;*TEST 526      BIS TEST - SM0,DM1 - N:C = 1000
10266                          ;;*****************************************************************
10267  035430                  TST526:
10268  035430  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10269  035432  012700  000526          MOV     #526,R0         ;;LOAD R0 WITH TEST NUMBER
10270  035436  013701  035460          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10271  035442  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10272  035446  005004                  CLR     R4              ;RESULT S / B = 000000
10273  035450  005005                  CLR     R5              ;SRC OPR = 000000
10274  035452  005012                  CLR     (R2)            ;[DEST] = 000000
10275  035454  000257                  CCC                     ;CLEAR FLAGS
10276  035456  000270                  SEN                     ;N:C = 1000
10277
10278  035460  050512          2$:     BIS     R5,(R2)         ;TEST THE BIS
10279
10280  035462  100403                  BMI     3$              ;N:C = 0100
10281  035464  001002                  BNE     3$
10282  035466  102401                  BVS     3$
10283  035470  103001                  BCC     4$
10284
10285  035472  104001          3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10286
10287  035474  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10288  035476  001402                  BEQ     TST527          ;;BR IF YES
10289
10290  035500  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10291  035502  104001          5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10292
10293                          ;;*****************************************************************
10294                          ;*TEST 527      BIC TEST - SM0,DM1 - N:C - 0111
10295                          ;;*****************************************************************
10296  035504                  TST527:
10297  035504  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10298  035506  012700  000527          MOV     #527,R0         ;;LOAD R0 WITH TEST NUMBER
10299  035512  013701  035542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10300  035516  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10301  035522  012704  100000          MOV     #100000,R4      ;RESULT S / B = 100000
10302  035526  012705  077777          MOV     #77777,R5               ;SRC OPR - 77777
10303  035532  012712  177777          MOV     #-1,(R2)        ;[DEST] = 177777
10304  035536  000257                  CCC                     ;CLEAR FLAGS
10305  035540  000267                  267                     ;N:C = 0111
10306
10307  035542  040512          2$:     BIC     R5,(R2)         ;TEST THE BIC
10308
10309  035544  100003                  BPL     3$              ;N:C - 1001
10310  035546  001402                  BEQ     3$
10311  035550  102401                  BVS     3$
10312  035552  103401                  BCS     4$
10313
10314  035554  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10315
```

```
10316  035556  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10317  035560  001402                          BEQ     TST530          ;;BR IF YES
10318
10319  035562  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10320  035564  104C01                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10321
10322                                  ;;**********************************************************
10323                                  ;*TEST 530       BIC TEST - SM0,DM1 - N:C = 1000
10324                                  ;;**********************************************************
10325  035566                          TST530:
10326  035566  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10327  035570  012700  000530                  MOV     #530,R0         ;;LOAD R0 WITH TEST NUMBER
10328  035574  013701  035616                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10329  035600  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10330  035604  005004                          CLR     R4              ;RESULT S / B = 000000
10331  035606  005005                          CLR     R5              ;SRC OPR = 000000
10332  035610  005012                          CLR     (R2)            ;[DEST] = 000000
10333  035612  000257                          CCC                     ;CLEAR FLAGS
10334  035614  000270                          SEN                     ;N:C = 1000
10335
10336  035616  040512                  2$:     BIC     R5,(R2)         ;TEST THE BIC
10337
10338  035620  100403                          BMI     3$              ;N:C = 0100
10339  035622  001002                          BNE     3$
10340  035624  102401                          BVS     3$
10341  035626  103001                          BCC     4$
10342
10343  035630  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10344
10345  035632  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10346  035634  001402                          BEQ     TST531          ;;BR IF YES
10347
10348  035636  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10349  035640  104001                  5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10350
10351                                  ;;**********************************************************
10352                                  ;*TEST 531       BIT TEST - SM0,DM1 - N:C = 0111
10353                                  ;;**********************************************************
10354  035642                          TST531:
10355  035642  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10356  035644  012700  000531                  MOV     #531,R0         ;;LOAD R0 WITH TEST NUMBER
10357  035650  013701  035700                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10358  035654  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10359  035660  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10360  035664  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10361  035670  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10362  035674  000257                          CCC                     ;CLEAR FLAGS
10363  035676  000267                          267                     ;N:C = 0111
10364
10365  035700  030512                  2$:     BIT     R5,(R2)         ;TEST THE BIT
10366
10367  035702  100003                          BPL     3$              ;N:C = 1001
10368  035704  001402                          BEQ     3$
10369  035706  102401                          BVS     3$
10370  035710  103401                          BCS     4$
10371
```

```
10372  035712  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10373
10374  035714  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10375  035716  001402                          BEQ     TST532          ;;BR IF YES
10376
10377  035720  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10378  035722  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10379
10380                                  ;;*****************************************************************
10381                                  ;*TEST 532       BIT TEST - SM0,DM1 - N:C = 1000
10382                                  ;;*****************************************************************
10383  035724                          TST532:
10384  035724  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10385  035726  012700  000532                  MOV     #532,R0         ;;LOAD R0 WITH TEST NUMBER
10386  035732  013701  035762                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10387  035736  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10388  035742  012704  052525                  MOV     #52525,R4       ;RESULT S / B = 52525
10389  035746  012705  125252                  MOV     #125252,R5      ;SRC OPR = 125252
10390  035752  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 52525
10391  035756  000257                          CCC                     ;CLEAR FLAGS
10392  035760  000270                          SEN                     ;N:C = 1000
10393
10394  035762  030512                  2$:     BIT     R5,(R2)         ;TEST THE BIT
10395
10396  035764  100403                          BMI     3$              ;N:C = 0100
10397  035766  001002                          BNE     3$
10398  035770  102401                          BVS     3$
10399  035772  103001                          BCC     4$
10400
10401  035774  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10402
10403  035776  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10404  036000  001402                          BEQ     TST533          ;;BR IF YES
10405
10406  036002  011203                          MOV     (R2),R3         ,GET THE WAS DATA
10407  036004  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10408                                  ;;*****************************************************************
10409                                  ;*TEST 533       CMP TEST - SM0,DM1 - N:C = 1010
10410                                  ;;*****************************************************************
10411  036006                          TST533:
10412  036006  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10413  036010  012700  000533                  MOV     #533,R0         ;;LOAD R0 WITH TEST NUMBER
10414  036014  013701  036044                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10415  036020  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10416  036024  012704  177777                  MOV     #-1,R4          ;RESULT S / B = -1
10417  036030  012705  177777                  MOV     #-1,R5          ;SRC OPR = 177777
10418  036034  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
10419  036040  000257                          CCC                     ;CLEAR FLAGS
10420  036042  000272                          272                     ;N:C = 1010
10421
10422  036044  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10423
10424  036046  100403                          BMI     3$              ;N:C = 0100
10425  036050  001002                          BNE     3$
10426  036052  102401                          BVS     3$
10427  036054  103001                          BCC     4$
```

```
10428
10429  036056  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10430
10431  036060  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10432  036062  001402                          BEQ     TST534          ;;BR IF YES
10433
10434  036064  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10435  036066  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10436
10437                                  ;;*********************************************************
10438                                  ;*TEST 534      CMP TEST - SM0,DM1 - N:C = 0110
10439                                  ;;*********************************************************
10440  036070                          TST534:
10441  036070  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10442  036072  012700  000534                  MOV     #534,R0         ;;LOAD R0 WITH TEST NUMBER
10443  036076  013701  036124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10444  036102  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10445  036106  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10446  036112  005005                          CLR     R5              ;SRC OPR = 000000
10447  036114  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10448  036120  000257                          CCC                     ;CLEAR FLAGS
10449  036122  000266                          266                     ;N:C = 0110
10450
10451  036124  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10452
10453  036126  100003                          BPL     3$              ;N:C = 1001
10454  036130  001402                          BEQ     3$
10455  036132  102401                          BVS     3$
10456  036134  103401                          BCS     4$
10457
10458  036136  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10459
10460  036140  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10461  036142  001402                          BEQ     TST535          ;;BR IF YES
10462
10463  036144  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10464  036146  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10465
10466                                  ;;*********************************************************
10467                                  ;*TEST 535      CMP TEST - SM0,DM1 - N:C = 0000
10468                                  ;;*********************************************************
10469  036150                          TST535:
10470  036150  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10471  036152  012700  000535                  MOV     #535,R0         ;;LOAD R0 WITH TEST NUMBER
10472  036156  013701  036204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10473  036162  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10474  036166  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10475  036172  012705  100000                  MOV     #100000,R5      ;SRC OPR = 100000
10476  036176  012712  000001                  MOV     #+1,(R2)        ;[DEST] = +1
10477  036202  000257                          CCC                     ;CLEAR FLAGS
10478
10479  036204  020512                  2$:     CMP     R5,(R2)         ;TEST THE CMP
10480
10481  036206  100403                          BMI     3$              ;N:C = 0010
10482  036210  001402                          BEQ     3$
10483  036212  102001                          BVC     3$
```

```
10484  036214  103001                  BCC     4$
10485
10486  036216  104001          3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10487
10488  036220  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10489  036222  001402                  BEQ     TST536          ;;BR IF YES
10490
10491  036224  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10492  036226  104001          5$:     ERROR   1               ;CMP DELIVERED A RESULT
10493
10494                          ;;**********************************************************************
10495                          ;*TEST 536       BIS TEST - SM1,DM0 - N:C = 0111
10496                          ;;**********************************************************************
10497  036230                  TST536:
10498  036230  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10499  036232  012700  000536          MOV     #536,R0         ;;LOAD R0 WITH TEST NUMBER
10500  036236  013701  036262          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10501  036242  012704  177777          MOV     #-1,R4          ;RESULT S / B = 177777
10502  036246  012705  063332          MOV     #DWTA+10,R5     ;SRC ADDR = DWTA+10
10503  036252  012703  052525          MOV     #52525,R3       ;[DEST] = 52525
10504  036256  000257                  CCC                     ;CLEAR FLAGS
10505  036260  000267                  267                     ;N:C = 0111
10506
10507  036262  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10508
10509  036264  100003                  BPL     3$              ;N:C = 1001
10510  036266  001402                  BEQ     3$
10511  036270  102401                  BVS     3$
10512  036272  103401                  BCS     4$
10513
10514  036274  104002          3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10515
10516  036276  020403          4$:     CMP     R4,R3           ;CORRECT RESULT ?
10517  036300  001401                  BEQ     TST537          ;;BR IF YES
10518
10519  036302  104002          5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10520
10521                          ;;**********************************************************************
10522                          ;*TEST 537       BIS TEST - SM1,DM0 - N:C = 1000
10523                          ;;**********************************************************************
10524  036304                  TST537:
10525  036304  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10526  036306  012700  000537          MOV     #537,R0         ;;LOAD R0 WITH TEST NUMBER
10527  036312  013701  036332          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10528  036316  005004                  CLR     R4              ;RESULT S / B =000000
10529  036320  012705  063322          MOV     #DWTA,R5        ;SRC ADDR = DWTA
10530  036324  005003                  CLR     R3              ;[DEST] = 000000
10531  036326  000257                  CCC                     ;CLEAR FLAGS
10532  036330  000270                  SEN                     ;N:C = 1000
10533
10534  036332  051503          2$:     BIS     (R5),R3         ;TEST THE BIS
10535
10536  036334  100403                  BMI     3$              ;N:C = 0100
10537  036336  001002                  BNE     3$
10538  036340  102401                  BVS     3$
10539  036342  103001                  BCC     4$
```

```
10540
10541  036344  104002                  3$:     ERROR   2               ;BIS FAILED TO ALTER CODES PROPERLY
10542
10543  036346  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10544  036350  001401                          BEQ     TST540          ;;BR IF YES
10545
10546  036352  104002                  5$:     ERROR   2               ;BIS DELIVERED THE WRONG RESULT
10547
10548                                  ;;*******************************************************************
10549                                  ;*TEST 540       BIC TEST - SM1,DM0 - N:C = 0111
10550                                  ;;*******************************************************************
10551  036354                          TST540:
10552  036354  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10553  036356  012700  000540                  MOV     #540,R0         ;;LOAD R0 WITH TEST NUMBER
10554  036362  013701  036412                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10555  036366  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10556  036372  012705  063316                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10557  036376  012703  177777                  MOV     #-1,R3  ;[DEST] = 177777
10558  036402  012715  077777                  MOV     #77777,(R5)     ;SRC OPR = 77777
10559  036406  000257                          CCC                     ;CLEAR FLAGS
10560  036410  000267                          267                     ;N:C = 011'
10561
10562  036412  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10563
10564  036414  100003                          BPL     3$              ;N:C = 1001 ?
10565  036416  001402                          BEQ     3$
10566  036420  102401                          BVS     3$
10567  036422  103401                          BCS     4$
10568
10569  036424  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10570
10571  036426  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10572  036430  001401                          BEQ     TST541          ;;BR IF YES
10573
10574  036432  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10575
10576                                  ;;*******************************************************************
10577                                  ;*TEST 541       BIC TEST - SM1,DM0 - N:C = 1000
10578                                  ;;*******************************************************************
10579  036434                          TST541:
10580  036434  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10581  036436  012700  000541                  MOV     #541,R0         ;;LOAD R0 WITH TEST NUMBER
10582  036442  013701  036462                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10583  036446  005004                          CLR     R4              ;RESULT S / B = 000000
10584  036450  012705  063322                  MOV     #DWTA,R5        ;SRC ADDR = DWTA
10585  036454  005003                          CLR     R3              ;[DEST] =000000
10586  036456  000257                          CCC                     ;CLEAR FLAGS
10587  036460  000270                          SEN                     ;N:C = 1000
10588
10589  036462  041503                  2$:     BIC     (R5),R3         ;TEST THE BIC
10590
10591  036464  100403                          BMI     3$              ;N:C = 0100
10592  036466  001002                          BNE     3$
10593  036470  102401                          BVS     3$
10594  036472  103001                          BCC     4$
10595
```

```
10596  036474  104002                  3$:     ERROR   2               ;BIC FAILED TO ALTER CODES PROPERLY
10597
10598  036476  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10599  036500  001401                          BEQ     TST542          ;;BR IF YES
10600
10601  036502  104002                  5$:     ERROR   2               ;BIC DELIVERED THE WRONG RESULT
10602
10603                                  ;;**********************************************************
10604                                  ;*TEST 542       BIT TEST - SM1,DM0 - N:C = 0111
10605                                  ;;**********************************************************
10606  036504                          TST542:
10607  036504  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10608  036506  012700  000542                 MOV     #542,R0         ;;LOAD R0 WITH TEST NUMBER
10609  036512  013701  036534                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10610  036516  012704  100000                 MOV     #100000,R4      ;RESULT S / B = 100000
10611  036522  012705  063324                 MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10612  036526  010403                          MOV     R4,R3           ;[DEST] = 100000
10613  036530  000257                          CCC                     ;CLEAR FLAGS
10614  036532  000267                          267                     ;N:C = 0111
10615
10616  036534  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10617
10618  036536  100003                          BPL     3$              ;N:C = 1001 ?
10619  036540  001402                          BEQ     3$
10620  036542  102401                          BVS     3$
10621  036544  103401                          BCS     4$
10622
10623  036546  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10624
10625  036550  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10626  036552  001401                          BEQ     TST543          ;;BR IF YES
10627
10628  036554  104002                  5$:     ERROR   2               ;BIT DELIVERED A RESULT
10629
10630                                  ;;**********************************************************
10631                                  ;*TEST 543       BIT TEST - SM1,DM0 - N:C = 1000
10632                                  ;;**********************************************************
10633  036556                          TST543:
10634  036556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10635  036560  012700  000543                 MOV     #543,R0         ;;LOAD R0 WITH TEST NUMBER
10636  036564  013701  036606                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10637  036570  012704  052525                 MOV     #52525,R4       ;RESULT S / B = 52525
10638  036574  012705  063332                 MOV     #DWTA+10,R5     ;SRC ADDR - DWTA+10
10639  036600  010403                          MOV     R4,R3           ;[DEST] = 52525
10640  036602  000257                          CCC                     ;CLEAR FLAGS
10641  036604  000270                          SEN                     ;N:C = 1000
10642
10643  036606  031503                  2$:     BIT     (R5),R3         ;TEST THE BIT
10644
10645  036610  100403                          BMI     3$              ;N:C = 0100
10646  036612  001002                          BNE     3$
10647  036614  102401                          BVS     3$
10648  036616  103001                          BCC     4$
10649
10650  036620  104002                  3$:     ERROR   2               ;BIT FAILED TO ALTER CODES PROPERLY
10651
```

```
10652  036622  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10653  036624  001401                         BEQ     TST544          ;;BR IF YES
10654
10655  036626  104002                 5$:     ERROR   2               ;BIT DELIVERED A RESULT
10656                                 ;;*****************************************************************
10657                                 ;*TEST 544       CMP TEST - SM1,DM0 - N:C = 0110
10658                                 ;;*****************************************************************
10659  036630                         TST544:
10660  036630  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10661  036632  012700  000544                 MOV     #544,R0         ;;LOAD R0 WITH TEST NUMBER
10662  036636  013701  036660                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10663  036642  012704  000001                 MOV     #+1,R4          ;RESULT S / B = +1
10664  036646  012705  063322                 MOV     #DWTA,R5        ;SRC ADDR = DWTA
10665  036652  010403                         MOV     R4,R3           ;[DEST] = +1
10666  036654  000257                         CCC                     ;CLEAR FLAGS
10667  036656  000266                         266                     ;N:C = 0110
10668
10669  036660  021503                 2$:     CMP     (R5),R3         ;TEST THE CMP
10670
10671  036662  100003                         BPL     3$              ;N:C = 1001
10672  036664  001402                         BEQ     3$
10673  036666  102401                         BVS     3$
10674  036670  103401                         BCS     4$
10675
10676  036672  104002                 3$.     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10677
10678  036674  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10679  036676  001401                         BEQ     TST545          ;;BR IF YES
10680
10681  036700  104002                 5$:     ERROR   2               ;CMP DELIVERED A RESULT
10682
10683                                 ;;*****************************************************************
10684                                 ;*TEST 545       CMP TEST - SM1,DM0 - N:C = 1010
10685                                 ;;*****************************************************************
10686  036702                         TST545:
10687  036702  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10688  036704  012700  000545                 MOV     #545,R0         ;;LOAD R0 WITH TEST NUMBER
10689  036710  013701  036732                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10690  036714  012704  177777                 MOV     #-1,R4          ;RESULT S / B = 177777
10691  036720  012705  063324                 MOV     #DWTA+2,R5      ;SRC ADDR = DWTA+2
10692  036724  010403                         MOV     R4,R3           ;[DEST] = 177777
10693  036726  000257                         CCC                     ;CLEAR FLAGS
10694  036730  000272                         272                     ;N:C = 1010
10695
10696  036732  021503                 2$:     CMP     (R5),R3         ;TEST THE CMP
10697
10698  036734  100403                         BMI     3$              ;N:C = 0100
10699  036736  001002                         BNE     3$
10700  036740  102401                         BVS     3$
10701  036742  103001                         BCC     4$
10702
10703  036744  104002                 3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10704
10705  036746  020403                 4$:     CMP     R4,R3           ;CORRECT RESULT ?
10706  036750  001401                         BEQ     TST546          ;;BR IF YES
10707
```

```
10708  036752  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10709
10710                                  ;;*****************************************************************
10711                                  ;*TEST 546      CMP TEST - SM1,DM0 - N:C = 0000
10712                                  ;;*****************************************************************
10713  036754                          TST546:
10714  036754  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10715  036756  012700  000546                  MOV     #546,R0         ;;LOAD R0 WITH TEST NUMBER
10716  036762  013701  037010                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10717  036766  012704  000001                  MOV     #+1,R4          ;RESULT S / B = +1
10718  036772  012705  063316                  MOV     #MBUF1,R5       ;SRC ADDR = MBUF1
10719  036776  012703  000001                  MOV     #+1,R3          ;[DEST] = +1
10720  037002  012715  100000                  MOV     #100000,(R5)    ;SRC OPR = 100000
10721  037006  000257                          CCC                     ;CLEAR FLAGS
10722
10723  037010  021503                  2$:     CMP     (R5),R3         ;TEST THE CMP
10724
10725  037012  100403                          BMI     3$              ;N:C = 0010
10726  037014  001402                          BEQ     3$
10727  037016  102001                          BVC     3$
10728  037020  103001                          BCC     4$
10729
10730  037022  104002                  3$:     ERROR   2               ;CMP FAILED TO ALTER CODES PROPERLY
10731
10732  037024  020403                  4$:     CMP     R4,R3           ;CORRECT RESULT ?
10733  037026  001401                          BEQ     TST547          ;;BR IF YES
10734
10735  037030  104002                  5$:     ERROR   2               ;CMP DELIVERED A RESULT
10736
10737                                  ;;*****************************************************************
10738                                  ;*TEST 547      BIS  SM1,DM1 TEST - N:C = 0111
10739                                  ;;*****************************************************************
10740  037032                          TST547:
10741  037032  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10742  037034  012700  000547                  MOV     #547,R0         ;;LOAD R0 WITH TEST NUMBER
10743  037040  013701  037070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10744  037044  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10745  037050  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 1777777
10746  037054  012705  063332                  MOV     #DWTA+10,R5     ;SOURCE ADDR = DWTA+10
10747  037060  012712  052525                  MOV     #52525,(R2)     ;[DEST] = 052525
10748  037064  000257                          CCC                     ;CLEAR FLAGS
10749  037066  000267                          267                     ;N:C = 0111
10750
10751  037070  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10752
10753  037072  100003                          BPL     3$              ;N:C = 1001?
10754  037074  001402                          BEQ     3$
10755  037076  102401                          BVS     3$
10756  037100  103401                          BCS     4$
10757
10758  037102  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10759
10760  037104  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10761  037106  001402                          BEQ     TST550          ;;BR IF YES
10762
10763  037110  011203                          MOV     (R2),R3         ;GET THE WAS DATA
```

```
10764  037112  104 01                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10765
10766                                   ;;*****************************************************************
10767                                   ;*TEST 550       BIS  SM1,DM1 TEST - N:C = 1000
10768                                   ;;*****************************************************************
10769  037114                           TST550:
10770  037114  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10771  037116  012700  000550                  MOV     #550,R0         ;;LOAD R0 WITH TEST NUMBER
10772  037122  013701  037146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10773  037126  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10774  037132  005004                          CLR     R4              ;RESULT S / B = 000000
10775  037134  012705  063322                  MOV     #DWTA,R5        ;SOURCE ADDR = DWTA
10776  037140  005012                          CLR     (R2)            ;[DEST] = 000000
10777  037142  000257                          CCC                     ;CLEAR FLAGS
10778  037144  000270                          SEN                     ;N:C = 1000
10779
10780  037146  051512                  2$:     BIS     (R5),(R2)       ;TEST THE BIS
10781
10782  037150  100403                          BMI     3$              ;N:C - 0100 ?
10783  037152  001002                          BNE     3$
10784  037154  102401                          BVS     3$
10785  037156  103001                          BCC     4$
10786
10787  037160  104001                  3$:     ERROR   1               ;BIS FAILED TO ALTER CODES PROPERLY
10788
10789  037162  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10790  037164  001402                          BEQ     TST551          ;;BR IF YES
10791
10792  037166  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10793  037170  104001                  5$:     ERROR   1               ;BIS DELIVERED THE WRONG RESULT
10794
10795                                   ;;*****************************************************************
10796                                   ;*TEST 551       BIC  SM1,DM1 TEST - N:C = 0111
10797                                   ;;*****************************************************************
10798  037172                           TST551:
10799  037172  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10800  037174  012700  000551                  MOV     #551,R0         ;;LOAD R0 WITH TEST NUMBER
10801  037200  013701  037234                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10802  037204  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10803  037210  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10804  037214  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10805  037220  012715  077777                  MOV     #77777,(R5)     ;[SOURCE] = 77777
10806  037224  012712  177777                  MOV     #-1,(R2)        ;[DEST] = 177777
10807  037230  000257                          CCC                     ;CLEAR FLAGS
10808  037232  000267                          267                     ;N:C = 0111
10809
10810  037234  041512                  2$:     BIC     (R5),(R2)       ;TEST THE BIC
10811
10812  037236  100003                          BPL     3$              ;N:C = 1001 ?
10813  037240  001402                          BEQ     3$
10814  037242  102401                          BVS     3$
10815  037244  103401                          BCS     4$
10816
10817  037246  104001                  3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10818
10819  037250  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
```

```
10820  037252  001402                  BEQ     TST552          ;;BR IF YES
10821
10822  037254  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10823  037256  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10824
10825                          ;;**************************************************************
10826                          ;*TEST 552       BIC  SM1,DM1 TEST - N:C = 1000
10827                          ;;**************************************************************
10828  037260                  TST552:
10829  037260  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10830  037262  012700  000552          MOV     #552,R0         ;;LOAD R0 WITH TEST NUMBER
10831  037266  013701  037314          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10832  037272  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10833  037276  005004                  CLR     R4              ;RESULT S / B = 000000
10834  037300  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR - MBUF1
10835  037304  005015                  CLR     (R5)            ;[SOURCE] = 000000
10836  037306  005012                  CLR     (R2)            ;[DEST] = 000000
10837  037310  000257                  CCC                     ;CLEAR FLAGS
10838  037312  000270                  SEN                     ;N:C = 1000
10839
10840  037314  041512          2$:     BIC     (R5),(R2)       ;TEST THE BIC
10841
10842  037316  100403                  BMI     3$              ;N:C = 0100 ?
10843  037320  001002                  BNE     3$
10844  037322  102401                  BVS     3$
10845  037324  103001                  BCC     4$
10846
10847  037326  104001          3$:     ERROR   1               ;BIC FAILED TO ALTER CODES PROPERLY
10848
10849  037330  020412          4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10850  037332  001402                  BEQ     TST553          ;;BR IF YES
10851
10852  037334  011203                  MOV     (R2),R3         ;GET THE WAS DATA
10853  037336  104001          5$:     ERROR   1               ;BIC DELIVERED THE WRONG RESULT
10854
10855                          ;;**************************************************************
10856                          ;*TEST 553       BIT  SM1,DM1 TEST - N:C = 1000
10857                          ;;**************************************************************
10858  037340                  TST553:
10859  037340  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10860  037342  012700  000553          MOV     #553,R0         ;;LOAD R0 WITH TEST NUMBER
10861  037346  013701  037402          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10862  037352  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10863  037356  012704  125252          MOV     #125252,R4      ;RESULT S / B = 125252
10864  037362  012705  063316          MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10865  037366  012715  052525          MOV     #52525,(R5)     ;[SOURCE] = 052525
10866  037372  012712  125252          MOV     #125252,(R2)    ;[DEST] = 125252
10867  037376  000257                  CCC                     ;CLEAR FLAGS
10868  037400  000270                  SEN                     ;N:C = 1000
10869
10870  037402  031512          2$:     BIT     (R5),(R2)       ;TEST THE BIT
10871
10872  037404  100403                  BMI     3$              ;N:C = 0100 ?
10873  037406  001002                  BNE     3$
10874  037410  102401                  BVS     3$
10875  037412  103001                  BCC     4$
```

```
10876
10877  037414  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10878
10879  037416  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10880  037420  001402                          BEQ     TST554          ;;BR IF YES
10881
10882  037422  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10883  037424  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10884
10885                                  ;;*************************************************************
10886                                  ;*TEST 554       BIT  SM1,DM1 TEST - N:C = 0111
10887                                  ;;*************************************************************
10888  037426                          TST554:
10889  037426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10890  037430  012700  000554                  MOV     #554,R0         ;;LOAD R0 WITH TEST NUMBER
10891  037434  013701  037502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10892                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT11
10893  037440  032737  004000  063234          BIT     #BIT11,@#BPTLOC ;BREAKPOINT HALT SET ??
10894  037446  001401                          BEQ     .+4             ;BR IF NOT
10895  037450  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
10896  037452  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10897  037456  012704  100000                  MOV     #100000,R4      ;RESULT S / B = 100000
10898  037462  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10899  037466  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 100000
10900  037472  012712  100000                  MOV     #100000,(R2)    ;[DEST] = 100000
10901  037476  000257                          CCC                     ;CLEAR FLAGS
10902  037500  000267                          267                     ;N:C = 0111
10903
10904  037502  031512                  2$:     BIT     (R5),(R2)       ;TEST THE BIT
10905
10906  037504  100003                          BPL     3$              ;N:C = 1001 ?
10907  037506  001402                          BEQ     3$
10908  037510  102401                          BVS     3$
10909  037512  103401                          BCS     4$
10910
10911  037514  104001                  3$:     ERROR   1               ;BIT FAILED TO ALTER CODES PROPERLY
10912
10913  037516  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10914  037520  001402                          BEQ     TST555          ;;BR IF YES
10915
10916  037522  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10917  037524  104001                  5$:     ERROR   1               ;BIT DELIVERED A RESULT
10918
10919                                  ;;*************************************************************
10920                                  ;*TEST 555       CMP  SM1,DM1 TEST - N:C = 1000
10921                                  ;;*************************************************************
10922  037526                          TST555:
10923  037526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10924  037530  012700  000555                  MOV     #555,R0         ;;LOAD R0 WITH TEST NUMBER
10925  037534  013701  037566                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10926  037540  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10927  037544  012704  177777                  MOV     #-1,R4          ;RESULT S / B = 177777
10928  037550  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10929  037554  012715  177777                  MOV     #-1,(R5)        ;[SOURCE] = 177777
10930  037560  010412                          MOV     R4,(R2)         ;[DEST] = 177777
10931  037562  000257                          CCC                     ;CLEAR FLAGS
```

```
10932  037564  000272                          272                     ;N:C = 1010
10933
10934  037566  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
10935
10936  037570  100403                          BMI     3$              ;N:C = 0100 ?
10937  037572  001002                          BNE     3$
10938  037574  102401                          BVS     3$
10939  037576  103001                          BCC     4$
10940
10941  037600  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10942
10943  037602  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10944  037604  001402                          BEQ     TST556          ;;BR IF YES
10945
10946  037606  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10947  037610  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10948
10949                                  ;;**********************************************************
10950                                  ;*TEST 556       CMP  SM1,DM1 TEST - N:C = 0110
10951                                  ;;**********************************************************
10952  037612                          TST556:
10953  037612  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10954  037614  012700  000556                  MOV     #556,R0         ;;LOAD R0 WITH TEST NUMBER
10955  037620  013701  037652                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10956  037624  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10957  037630  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
10958  037634  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
10959  037640  005015                          CLR     (R5)            ;[SOURCE] = 000000
10960  037642  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
10961  037646  000257                          CCC                     ;CLEAR FLAGS
10962  037650  000266                          266                     ,N:C = 0110
10963
10964  037652  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
10965
10966  037654  100003                          BPL     3$              ;N:C = 1001 ?
10967  037656  001402                          BEQ     3$
10968  037660  102401                          BVS     3$
10969  037662  103401                          BCS     4$
10970
10971  037664  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
10972
10973  037666  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
10974  037670  001402                          BEQ     TST557          ;;BR IF YES
10975
10976  037672  011203                          MOV     (R2),R3         ;GET THE WAS DATA
10977  037674  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
10978
10979                                  ;;**********************************************************
10980                                  ;*TEST 557       CMP  SM1,DM1 TEST - N:C = 0000
10981                                  ;;**********************************************************
10982  037676                          TST557:
10983  037676  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
10984  037700  012700  000557                  MOV     #557,R0         ;;LOAD R0 WITH TEST NUMBER
10985  037704  013701  037736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
10986  037710  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
10987  037714  012704  000001                  MOV     #+1,R4          ;RESULT S / B = 000001
```

```
 10988  037720  012705  063316                  MOV     #MBUF1,R5       ;SOURCE ADDR = MBUF1
 10989  037724  012715  100000                  MOV     #100000,(R5)    ;[SOURCE] = 000000
 10990  037730  012712  000001                  MOV     #+1,(R2)        ;[DEST] = 000001
 10991  037734  000257                          CCC                     ;CLEAR FLAGS
 10992
 10993  037736  021512                  2$:     CMP     (R5),(R2)       ;TEST THE CMP
 10994
 10995  037740  100403                          BMI     3$              ;N:C = 0010 ?
 10996  037742  001402                          BEQ     3$
 10997  037744  102001                          BVC     3$
 10998  037746  103001                          BCC     4$
 10999
 11000  037750  104001                  3$:     ERROR   1               ;CMP FAILED TO ALTER CODES PROPERLY
 11001
 11002  037752  020412                  4$:     CMP     R4,(R2)         ;CORRECT RESULT ?
 11003  037754  001402                          BEQ     TST560          ;;BR IF YES
 11004
 11005  037756  011203                          MOV     (R2),R3         ;GET THE WAS DATA
 11006  037760  104001                  5$:     ERROR   1               ;CMP DELIVERED A RESULT
 11007
 11008                                  ;;*******************************************************
 11009                                  ;*TEST 560      BISB  SM1,DM0 TEST - SOURCE ADDR ODD
 11010                                  ;;*******************************************************
 11011  037762                          TST560:
 11012  037762  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11013  037764  012700  000560                  MOV     #560,R0         ;;LOAD R0 WITH TEST NUMBER
 11014  037770  013701  040010                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11015  037774  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 11016  040000  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 11017  040004  005003                          CLR     R3              ;[DEST] = 000000
 11018  040006  000257                          CCC                     ;SCOPE SYNC
 11019
 11020  040010  151503                  2$:     BISB    (R5),R3         ;TEST THE BISB
 11021
 11022  040012  020403                          CMP     R4,R3           ;RESULT CORRECT ?
 11023  040014  001401                          BEQ     TST561          ;;BR IF YES
 11024
 11025  040016  104002                  3$:     ERROR   2               ;BISB DELIVERED THE WRONG RESULT
 11026
 11027                                  ;;*******************************************************
 11028                                  ;*TEST 561      BISB  SM1,DM1 TEST - SOURCE ADDR ODD
 11029                                  ;;*******************************************************
 11030  040020                          TST561:
 11031  040020  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11032  040022  012700  000561                  MOV     #561,R0         ;;LOAD R0 WITH TEST NUMBER
 11033  040026  013701  040052                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11034  040032  012702  063312                  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
 11035  040036  012704  000377                  MOV     #377,R4         ;RESULT S / B = 377
 11036  040042  012705  064631                  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
 11037  040046  005012                          CLR     (R2)            ;[DEST] = 000000
 11038  040050  000257                          CCC                     ;SCOPE SYNC
 11039
 11040  040052  151512                  2$:     BISB    (R5),(R2)       ;TEST THE BISB
 11041
 11042  040054  020412                          CMP     R4,(R2)         ;CORRECT RESULT
 11043  040056  001402                          BEQ     TST562          ;;BR IF YES
```

```
11044
11045  040060  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11046  040062  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11047
11048                                ;;*****************************************************************
11049                                ;*TEST 562      BISB  SM1,DM2 TEST - SOURCE ADDR ODD
11050                                ;;*****************************************************************
11051  040064                        TST562:
11052  040064  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11053  040066  012700  000562                MOV     #562,R0         ;;LOAD R0 WITH TEST NUMBER
11054  040072  013701  040120                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11055  040076  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11056  040102  012704  000377                MOV     #377,R4         ;RESULT S / B - 377
11057  040106  012705  064631                MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11058  040112  005012                        CLR     (R2)            ;[DEST] = 000000
11059  040114  010203                        MOV     R2,R3           ;DEST ADDR IN R3
11060  040116  000257                        CCC                     ;SCOPE SYNC
11061
11062  040120  151523                2$:     BISB    (R5),(R3)+      ;TEST THE BISB
11063
11064  040122  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11065  040124  001402                        BEQ     TST563          ;;BR IF YES
11066
11067  040126  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11068  040130  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11069
11070                                ;;*****************************************************************
11071                                ;*TEST 563      BISB  SM1,DM3 TEST - SOURCE ADDR ODD
11072                                ;;*****************************************************************
11073  040132                        TST563:
11074  040132  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11075  040134  012700  000563                MOV     #563,R0         ;;LOAD R0 WITH TEST NUMBER
11076  040140  013701  040170                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11077  040144  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11078  040150  012704  000377                MOV     #377,R4         ;RESULT S / B = 377
11079  040154  012705  064631                MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11080  040160  005012                        CLR     (R2)            ;[DEST] = 000000
11081  040162  012703  063306                MOV     #ATA+10,R3      ;BASE DEST ADDR = ATA+10
11082  040166  000257                        CCC                     ;SCOPE SYNC
11083
11084  040170  151533                2$:     BISB    (R5),@(R3)+     ;TEST THE BISB
11085
11086  040172  020412                        CMP     R4,(R2)         ;CORRECT RESULT
11087  040174  001402                        BEQ     TST564          ;;BR IF YES
11088
11089  040176  011203                        MOV     (R2),R3         ;GET THE WAS DATA
11090  040200  104001                3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11091
11092                                ;;*****************************************************************
11093                                ;*TEST 564      BISB  SM1,DM4 TEST - SOURCE ADDR ODD
11094                                ;;*****************************************************************
11095  040202                        TST564:
11096  040202  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11097  040204  012700  000564                MOV     #564,R0         ;;LOAD R0 WITH TEST NUMBER
11098  040210  013701  040240                MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11099  040214  012702  063312                MOV     #MBUF0,R2       ;DEST ADDR - MBUF0
```

```
11100  040220  012704  177400          MOV     #177400,R4      ;RESULT S / B = 177400
11101  040224  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11102  040230  012703  063314          MOV     #MBUF0+2,R3     ;BASE DEST ADDR - MBUF0+2
11103  040234  005012                  CLR     (R2)            ;[DEST] = 000000
11104  040236  000257                  CCC                     ;SCOPE SYNC
11105
11106  040240  151543          2$:     BISB    (R5),-(R3)      ;TEST THE BISB
11107
11108  040242  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11109  040244  001402                  BEQ     TST565          ;;BR IF YES
11110
11111  040246  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11112  040250  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11113
11114                          ;;*****************************************************************
11115                          ;*TEST 565       BISB  SM1,DM5 TEST - SOURCE ADDR ODD
11116                          ;;*****************************************************************
11117  040252                  TST565:
11118  040252  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11119  040254  012700  000565          MOV     #565,R0         ;;LOAD R0 WITH TEST NUMBER
11120  040260  013701  040310          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11121  040264  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11122  040270  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11123  040274  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11124  040300  012703  063310          MOV     #ATA+12,R3      ;BASE DEST ADDR = ATA+12
11125  040304  005012                  CLR     (R2)            ;[DEST] = 000000
11126  040306  000257                  CCC                     ;SCOPE SYNC
11127
11128  040310  151553          2$:     BISB    (R5),@-(R3)     ;TEST THE BISB
11129
11130  040312  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11131  040314  001402                  BEQ     TST566          ;;BR IF YES
11132
11133  040316  011203                  MOV     (R2),R3         ;GET THE WAS DATA
11134  040320  104001          3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11135
11136                          ;;*****************************************************************
11137                          ;*TEST 566       BISB  SM1,DM6 TEST - SOURCE ADDR ODD
11138                          ;;*****************************************************************
11139  040322                  TST566:
11140  040322  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11141  040324  012700  000566          MOV     #566,R0         ;;LOAD R0 WITH TEST NUMBER
11142  040330  013701  040360          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11143  040334  012702  063312          MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11144  040340  012704  000377          MOV     #377,R4         ;RESULT S / B = 377
11145  040344  012705  064631          MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11146  040350  012703  063320          MOV     #MBUF0+6,R3     ;BASE DEST ADDR - MBUF0+6
11147  040354  005012                  CLR     (R2)            ;[DEST] = 000000
11148  040356  000257                  CCC                     ;SCOPE SYNC
11149
11150  040360  151563  177772  2$:     BISB    (R5),-6(R3)     ;TEST THE BISB
11151
11152  040364  020412                  CMP     R4,(R2)         ;CORRECT RESULT
11153  040366  001402                  BEQ     TST567          ;;BR IF YES
11154
11155  040370  011203                  MOV     (R2),R3         ;GET THE WAS DATA
```

```
11156  040372  104001                    3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11157
11158                                    ;;*****************************************************************
11159                                    ;*TEST 567      BISB  SM1,DM7 TEST - SOURCE ADDR ODD
11160                                    ;;*****************************************************************
11161  040374                            TST567:
11162  040374  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11163  040376  012700  000567                   MOV     #567,R0         ;;LOAD R0 WITH TEST NUMBER
11164  040402  013701  040432                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11165  040406  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11166  040412  012704  000377                   MOV     #377,R4         ;RESULT S / B = 377
11167  040416  012705  064631                   MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
11168  040422  012703  063276                   MOV     #ATA,R3         ;BASE DEST ADDR = ATA
11169  040426  005012                           CLR     (R2)            ;[DEST] = 000000
11170  040430  000257                           CCC                     ;SCOPE SYNC
11171
11172  040432  151573  000010           2$:     BISB    (R5),@10(R3)    ;TEST THE BISB
11173
11174  040436  020412                           CMP     R4,(R2)         ;CORRECT RESULT
11175  040440  001402                           BEQ     TST570          ;;BR IF YES
11176
11177  040442  011203                           MOV     (R2),R3         ;GET THE WAS DATA
11178  040444  104001                    3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11179
11180                                    ;;*****************************************************************
11181                                    ;*TEST 570      BISB  SM0,DM2 TEST - DEST ADDR EVEN
11182                                    ;;*****************************************************************
11183  040446                            TST570:
11184  040446  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11185  040450  012700  000570                   MOV     #570,R0         ;;LOAD R0 WITH TEST NUMBER
11186  040454  013701  040476                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11187  040460  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11188  040464  012704  000377                   MOV     #377,R4         ;RESULT S / B = 377
11189  040470  010203                           MOV     R2,R3           ;DEST ADDR IN R3
11190  040472  005012                           CLR     (R2)            ;[DEST] = 000000
11191  040474  000257                           CCC                     ;SCOPE SYNC
11192
11193  040476  150423                    2$:     BISB    R4,(R3)+        ;TEST THE BISB
11194
11195  040500  020412                           CMP     R4,(R2)         ;CORRECT RESULT
11196  040502  001402                           BEQ     TST571          ;;BR IF YES
11197
11198  040504  011203                           MOV     (R2),R3         ;GET THE WAS DATA
11199  040506  104001                    3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT
11200
11201                                    ;;*****************************************************************
11202                                    ;*TEST 571      BISB  SM0,DM1 TEST - DEST ADDR ODD
11203                                    ;;*****************************************************************
11204  040510                            TST571:
11205  040510  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11206  040512  012700  000571                   MOV     #571,R0         ;;LOAD R0 WITH TEST NUMBER
11207  040516  013701  040546                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11208  040522  012702  063312                   MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
11209  040526  012704  177400                   MOV     #177400,R4      ;RESULT S / B = 177400
11210  040532  012705  000377                   MOV     #377,R5         ;[R5]=SOURCE OPR = 377
11211  040536  012703  063313                   MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
```

```
                        CLR     (R2)            ;[DEST] = 000000
                        CCC                     ;SCOPE SYNC

040546  150513  2$:     BISB    R5,(R3)         ;TEST THE BISB

040550  020412          CMP     R4,(R2)         ;CORRECT RESULT
040552  001402          BEQ     TST572          ;;BR IF YES

040554  011203          MOV     (R2),R3         ;GET THE WAS DATA
040556  104001  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT

                        ;;*******************************************************
                        ;*TEST 572      BISB  SM0,DM1 TEST - DEST ADDR EVEN
                        ;;*******************************************************
040560          TST572:
040560  000004          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
040562  012700  000572  MOV     #572,R0         ;;LOAD R0 WITH TEST NUMBER
040566  013701  040610  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
040572  012702  063312  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
040576  012704  000377  MOV     #377,R4         ;RESULT S / B = 377
040602  010203          MOV     R2,R3           ;DEST ADDR IN R3
040604  005012          CLR     (R2)            ;[DEST] = 000000
040606  000257          CCC                     ;SCOPE SYNC

040610  150413  2$:     BISB    R4,(R3)         ;TEST THE BISB

040612  020412          CMP     R4,(R2)         ;CORRECT RESULT
040614  001402          BEQ     TST573          ;;BR IF YES

040616  011203          MOV     (R2),R3         ;GET THE WAS DATA
040620  104001  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESUL

                        ;;*******************************************************
                        ;*TEST 573      BISB  SM1,DM1 TEST - DEST ADDR ODD
                        ;;*******************************************************
040622          TST573:
040622  000004          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
040624  012700  000573  MOV     #573,R0         ;;LOAD R0 WITH TEST NUMBER
040630  013701  040660  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
040634  012702  063312  MOV     #MBUF0,R2       ;DEST ADDR = MBUF0
040640  012704  177400  MOV     #177400,R4      ;RESULT S / B = 177400
040644  012705  064631  MOV     #DBTA+1,R5      ;SOURCE ADDR = DBTA+1
040650  012703  063313  MOV     #MBUF0+1,R3     ;ODD DEST ADDR IN R3
040654  005012          CLR     (R2)            ;[DEST] = 000000
040656  000257          CCC                     ;SCOPE SYNC

040660  151513  2$:     BISB    (R5),(R3)       ;TEST THE BISB

040662  020412          CMP     R4,(R2)         ;CORRECT RESULT
040664  001402          BEQ     TST574          ;;BR IF YES

040666  011203          MOV     (R2),R3         ;GET THE WAS DATA
040670  104001  3$:     ERROR   1               ;BISB DELIVERED THE WRONG RESULT

                        ;;*******************************************************
                        ;*TEST 574      JMP MODE 1 TEST, FLAGS = 1111
```

```
11268
11269  040672
11270  040672  000004                  ;;*********************************************************************
                                       TST574:
                                               SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11271  040674  012700  000574                  MOV     #574,R0         ;;LOAD R0 WITH TEST NUMBER
11272  040700  013701  040712                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11273  040704  012702  040720                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11274  040710  000277                          SCC                     ;MAKE N:C = 1111
11275
11276  040712  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11277
11278  040714  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11279  040716  000405                          BR      TST575          ;;GO CALL SCOPE
11280
11281  040720  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11282  040722  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11283  040724  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11284  040726  100401                          BMI     TST575          ;;BR IF "N" STILL SET
11285
11286  040730  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED ONE
11287
11288                                  ;;*********************************************************************
11289                                  ;*TEST 575      JMP MODE 1 TEST, FLAGS = 0000
11290                                  ;;*********************************************************************
11291  040732                          TST575:
11292  040732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11293  040734  012700  000575                  MOV     #575,R0         ;;LOAD R0 WITH TEST NUMBER
11294  040740  013701  040752                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11295  040744  012702  040760                  MOV     #4$,R2          ;R2 CONTAINS JMP ADDRESS
11296  040750  000257                          CCC                     ;MAKE N:C = 0000
11297
11298  040752  000112                  2$:     JMP     (R2)            ;TEST THE JMP - GO TO 4$
11299
11300  040754  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11301  040756  000405                          BR      TST576          ;;GO CALL SCOPE
11302
11303  040760  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11304  040762  102402                          BVS     5$              ;BR IF JMP SET "V"
11305  040764  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11306  040766  100001                          BPL     TST576          ;;BR IF "N" STILL CLEAR
11307
11308  040770  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET ONE
11309
11310                                  ;;*********************************************************************
11311                                  ;*TEST 576      JMP MODE 2 TEST; FLAGS = 1111
11312                                  ;;*********************************************************************
11313  040772                          TST576:
11314  040772  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11315  040774  012700  000576                  MOV     #576,R0         ;;LOAD R0 WITH TEST NUMBER
11316  041000  013701  041012                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11317  041004  012702  041020                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11318  041010  000277                          SCC                     ;SET N:C = 1111
11319
11320  041012  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11321
11322  041014  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11323  041016  000411                          BR      TST577          ;;GO TO SCOPE EXIT
```

```
11324
11325  041020  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11326  041022  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11327  041024  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11328  041026  100401                          BMI     6$              ;BR IF "N" STILL SET
11329
11330  041030  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11331
11332  041032  022702  041022          6$:     CMP     #4$+2,R2        ;DID R2 GET AUTO-INCREMENTED?
11333  041036  001401                          BEQ     TST577          ;;BR IF YES
11334
11335  041040  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER (R2)
11336
11337                                  ;;*********************************************************************
11338                                  ;*TEST 577       JMP MODE 2 TEST; FLAGS = 0000
11339                                  ;;*********************************************************************
11340  041042                          TST577:
11341  041042  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11342  041044  012700  000577                  MOV     #577,R0         ;;LOAD R0 WITH TEST NUMBER
11343  041050  013701  041062                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11344  041054  012702  041070                  MOV     #4$,R2          ;R2 CONTAINS JUMP ADDRESS
11345  041060  000257                          CCC                     ;MAKE N:C = 0000
11346
11347  041062  000122                  2$:     JMP     (R2)+           ;TEST THE JMP - GO TO 4$
11348
11349  041064  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11350  041066  000405                          BR      TST600          ;;GO TO SCOPE EXIT
11351
11352  041070  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
11353  041072  102402                          BVS     5$              ;BR IF JMP SET "V"
11354  041074  001401                          BEQ     5$              ;BR IF JMP SET "Z"
11355  041076  100001                          BPL     TST600          ;;BR IF "N" IS CLEAR
11356
11357  041100  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
11358
11359                                  ;;*********************************************************************
11360                                  ;*TEST 600       JMP TEST MODE 3; FLAGS = 1111
11361                                  ;;*********************************************************************
11362  041102                          TST600:
11363  041102  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11364  041104  012700  000600                  MOV     #600,R0         ;;LOAD R0 WITH TEST NUMBER
11365  041110  013701  041122                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11366  041114  012702  041154                  MOV     #7$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
11367  041120  000277                          SCC                     ;SET N:C = 1111
11368
11369  041122  000132                  2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
11370
11371  041124  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11372  041126  000414                          BR      TST601          ;;GO TO SCOPE EXIT
11373
11374  041130  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
11375  041132  102002                          BVC     5$              ;BR IF JMP CLEARED "V"
11376  041134  001001                          BNE     5$              ;BR IF JMP CLEARED "Z"
11377  041136  100401                          BMI     6$              ;BR IF "N" STILL SET
11378
11379  041140  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEAR
```

```
 11380
 11381  041142  022702  041156          6$:     CMP     #7$+2,R2        ;DID JMP UPDATE R2?
 11382  041146  001404                          BEQ     TST601          ;;BR IF YES
 11383
 11384  041150  104006                          ERROR   6               ;JMP FAILED TO UPDATE REGISTER
```

```
 11385  041152  000402                    BR      TST601          ;;GO TO SCOPE EXIT
 11386  041154  041130            7$:     4$                      ;JMP3 CONTAINS JUMP ADDRESS
 11387  041156  104006                    ERROR   6               ;ERROR CALL OCCURS IF MODE3 HAPPENS
 11388                                                           ;TO EXECUTE AS MODE 1 OR 2 AND
 11389                                                           ;4$ IS LEGAL INSTRUCTION
 11390
 11391                            ;;*****************************************************************
 11392                            ;*TEST 601      JMP TEST MODE 3; FLAGS = 0000
 11393                            ;;*****************************************************************
 11394  041160                    TST601:
 11395  041160  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11396  041162  012700  000601            MOV     #601,R0         ;;LOAD R0 WITH TEST NUMBER
 11397  041166  013701  041200            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11398  041172  012702  041222            MOV     #6$,R2          ;R2 CONTAINS ADDRESS OF JUMP ADDRESS
 11399  041176  000257                    CCC                     ;MAKE N:C = 0000
 11400
 11401  041200  000132            2$:     JMP     @(R2)+          ;TEST THE JMP - GO TO 4$
 11402
 11403  041202  104006            3$:     ERROR   6               ;JMP FAILED TO LOAD THE PC
 11404  041204  000410                    BR      TST602          ;;GO TO SCOPE EXIT
 11405
 11406  041206  103403            4$:     BCS     5$              ;BR IF JMP SET "C"
 11407  041210  102402                    BVS     5$              ;BR IF JMP SET "V"
 11408  041212  001401                    BEQ     5$              ;BR IF JMP SET "Z"
 11409  041214  100004                    BPL     TST602          ;;BR IF "N" STILL CLEAR
 11410
 11411  041216  104006            5$:     ERROR   6               ;JMP ALTERED CODES - SET
 11412  041220  000402                    BR      TST602          ;;GO TO SCOPE EXIT
 11413
 11414  041222  041206            6$:     4$                      ;JUMP ADDRESS IN 6$
 11415  041224  104006                    ERROR   6               ;JMP MODE 3 EXECUTED LIKE MODE   OR 2
 11416
 11417                            ;;*****************************************************************
 11418                            ;*TEST 602      JMP TEST MODE 4; FLAGS = 1111
 11419                            ;;*****************************************************************
 11420  041226                    TST602:
 11421  041226  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11422  041230  012700  000602            MOV     #602,R0         ;;LOAD R0 WITH TEST NUMBER
 11423  041234  013701  041246            MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11424  041240  012702  041256            MOV     #5$,R2  ;[R2] = JMP ADDRESS PLUS 2
 11425  041244  000277                    SCC                     ;MAKE N:C = 1111
 11426
 11427  041246  000142            2$:     JMP     -(R2)           ;TEST THE JMP - GO TO 5$ MINUS 2
 11428
 11429  041250  104006            3$:     ERROR   6               ;JMP FAILED TO LOAD PC
 11430  041252  000414                    BR      TST603          ;;GO TO SCOPE EXIT
 11431
 11432  041254  000402                    BR      4$              ;GO TEST FLAGS - JMP LOADED PC OK
 11433  041256  104006            5$:     ERROR   6               ;JMP FAILED TO AUTO-DECREMENT R2
 11434  041260  000411                    BR      TST603          ;;GO TO SCOPE EXIT
 11435
 11436  041262  103003            4$:     BCC     7$              ;BR IF JMP CLEARED "C"
 11437  041264  102002                    BVC     7$              ;BR IF JMP CLEARED "V"
 11438  041266  001001                    BNE     7$              ;BR IF JMP CLEARED "Z"
 11439  041270  100401                    BMI     6$              ;BR IF "N" STILL SET
 11440
```

```
 11441  041272  104006                  7$:     ERROR   6               ;JMP ALTERED FLAGS
 11442
 11443  041274  022702  041254          6$:     CMP     #5$-2,R2        ;DID JMP UPDATE R2 PROPERLY?
 11444  041300  001401                          BEQ     TST603          ;;BR IF YES
 11445
 11446  041302  104006                  9$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
 11447
 11448                                  ;;*******************************************************************
 11449                                  ;*TEST 603      JMP TEST MODE 4; FLAGS = 0000
 11450                                  ;;*******************************************************************
 11451  041304                          TST603:
 11452  041304  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11453  041306  012700  000603                  MOV     #603,R0         ;;LOAD R0 WITH TEST NUMBER
 11454  041312  013701  041324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11455  041316  012702  041334                  MOV     #4$+2,R2        ;[R2] = JUMP ADDRESS PLUS 2
 11456  041322  000257                          CCC                     ;MAKE N:C = 0000
 11457
 11458  041324  000142                  2$:     JMP     -(R2)           ;TEST THE JMP - TO TO 4$
 11459
 11460  041326  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
 11461  041330  000405                          BR      TST604          ;;GO TO SCOPE EXIT
 11462
 11463  041332  103403                  4$:     BCS     5$              ;BR IF JMP SET "C"
 11464  041334  102402                          BVS     5$              ;BR IF JMP SET "V"
 11465  041336  001401                          BEQ     5$              ;BR IF JMP SET "Z"
 11466  041340  100001                          BPL     TST604          ;;BR IF "N" STILL CLEAR
 11467
 11468  041342  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - SET
 11469
 11470                                  ;;*******************************************************************
 11471                                  ;*TEST 604      JMP TEST MODE 5; FLAGS = 1111
 11472                                  ;;*******************************************************************
 11473  041344                          TST604:
 11474  041344  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11475  041346  012700  000604                  MOV     #604,R0         ;;LOAD R0 WITH TEST NUMBER
 11476  041352  013701  041364                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11477  041356  012702  041420                  MOV     #JMP5,R2        ;JMP CONTAINS ADDR+2 OF JUMP ADDRESS
 11478  041362  000277                          SCC
 11479
 11480  041364  000152                  2$:     JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
 11481
 11482  041366  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
 11483  041370  000414                          BR      TST605          ;;GO TO SCOPE EXIT
 11484
 11485  041372  103003                  4$:     BCC     5$              ;BR IF JMP CLEARED "C"
 11486  041374  102002                          BVC     5$
 11487  041376  001001                          BNE     5$
 11488  041400  100401                          BMI     6$
 11489
 11490  041402  104006                  5$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
 11491
 11492  041404  022702  041416          6$:     CMP     #JMP5-2,R2      ;DID R2 GET AUTO-DECREMENTED
 11493  041410  001404                          BEQ     TST605          ;;BR IF YES
 11494
 11495  041412  104006                  7$:     ERROR   6               ;JMP FAILED TO UPDATE REGISTER
 11496  041414  000402                          BR      TST605          ;;GO TO SCOPE EXIT
```

```
 11497  041416  041372                          4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
 11498  041420  104006                JMP5:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
 11499
 11500                                  ;;*****************************************************************
 11501                                  ;*TEST 605       JMP TEST MODE 5; FLAG = 0000
 11502                                  ;;*****************************************************************
 11503  041422                          TST605:
 11504  041422  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11505  041424  012700  000605                  MOV     #605,R0         ;;LOAD R0 WITH TEST NUMBER
 11506  041430  013701  041442                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11507  041434  012702  041466                  MOV     #JMP5A,R2       ;[R2] = ADDR +2 OF JUMP ADDRESS
 11508  041440  000257                          CCC                     ;SET N:C = 0000
 11509
 11510  041442  000152                2$:       JMP     @-(R2)          ;TEST THE JMP - GO TO 4$
 11511
 11512  041444  104006                3$:       ERROR   6               ;JMP FAILED TO LOAD PC
 11513  041446  000410                          BR      TST606          ;;GO TO SCOPE EXIT
 11514
 11515  041450  103403                4$:       BCS     5$              ;BR IF JMP SET "C"
 11516  041452  102402                          BVS     5$              ;BR IF JMP SET "V"
 11517  041454  001401                          BEQ     5$              ;BR IF JMP SET "Z"
 11518  041456  100004                          BPL     TST606          ;;BR IF "N" STILL CLEAR
 11519
 11520  041460  104006                5$:       ERROR   6               ;JMP ALTERED THE CODES - SET
 11521  041462  000402                          BR      TST606          ;;GO TO SCOPE EXIT
 11522
 11523  041464  041450                          4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
 11524  041466  104006                JMP5A:    ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2
 11525
 11526                                  ;;*****************************************************************
 11527                                  ;*TEST 606       JMP TEST MODE 6; FLAGS = 1111
 11528                                  ;;*****************************************************************
 11529  041470                          TST606:
 11530  041470  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 11531  041472  012700  000606                  MOV     #606,R0         ;;LOAD R0 WITH TEST NUMBER
 11532  041476  013701  041510                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
 11533  041502  012702  041534                  MOV     #7$,R2  ;[R2] = BASE ADDRESS TO BE INDEXED
 11534  041506  000277                          SCC                     ;MAKE N:C = 1111
 11535
 11536  041510  000162  177764        2$:       JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
 11537
 11538  041514  104006                3$:       ERROR   6               ;JMP FAILED TO LOAD THE PC
 11539  041516  000407                          BR      TST607          ;;GO TO SCOPE EXIT
 11540
 11541  041520  103003                4$:       BCC     5$              ;BR IF JMP CLEARED "C"
 11542  041522  102002                          BVC     5$
 11543  041524  001001                          BNE     5$
 11544  041526  100403                          BMI     TST607          ;;BR IF "N" STILL SET
 11545
 11546  041530  104006                5$:       ERROR   6               ;JMP ALTERED CODES - CLEARED
 11547  041532  000401                          BR      TST607          ;;GO TO SCOPE EXIT
 11548
 11549  041534  104006                7$:       ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2 OR
 11550                                                                  ;FAILED TO INDEX [R2]
 11551
 11552                                  ;;*****************************************************************
```

```
11553                                   ;*TEST 607      JMP TEST MODE 6; FLAGS = 0000
11554                                   ;;*****************************************************************
11555  041536                           TST607:
11556  041536  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11557  041540  012700  000607                  MOV     #607,R0         ;;LOAD R0 WITH TEST NUMBER
11558  041544  013701  041556                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11559  041550  012702  041602                  MOV     #7$,R2  ;[R2] = BASE ADDRESS FOR JUMP
11560  041554  000257                          CCC                     ;MAKE N:C = 0000
11561
11562  041556  000162  177764           2$:     JMP     4$-7$(R2)       ;TEST THE JMP - GO TO 4$
11563
11564  041562  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11565  041564  000407                          BR      TST610          ;;GO TO SCOPE EXIT
11566
11567  041566  103403                   4$:     BCS     5$              ;BR IF JMP SET "C"
11568  041570  102402                           BVS     5$              ;BR IF JMP SET "V"
11569  041572  001401                           BEQ     5$              ;BR IF JMP SET "Z"
11570  041574  100003                           BPL     TST610          ;;BR IF "N" STILL CLEAR
11571
11572  041576  104006                   5$:     ERROR   6               ;JMP ALTERED CODES
11573  041600  000401                          BR      TST610          ;;GO TO SCOPE EXIT
11574
11575  041602  104006                   7$:     ERROR   6               ;JMP EXECUTED LIKE A MODE 1 OR 2, OR
11576                                                                   ;FAILED TO INDEX [R2]
11577
11578                                   ;;*****************************************************************
11579                                   ;*TEST 610      JMP TEST MODE 7; FLAGS = 1111
11580                                   ;;*****************************************************************
11581  041604                           TST610:
11582  041604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11583  041606  012700  000610                  MOV     #610,R0         ;;LOAD R0 WITH TEST NUMBER
11584  041612  013701  041624                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11585  041616  012702  041634                  MOV     #5$,R2  ;[R2] = BASE ADDRESS
11586  041622  000277                          SCC                     ;MAKE N:C = 1111
11587
11588  041624  000172  000020           2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11589
11590  041630  104006                   3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11591  041632  000412                          BR      TST611          ;;GO TO SCOPE EXIT
11592
11593  041634  104006                   5$:     ERROR   6               ;JMP FAILED TO INDEX OR ACTED LIKE MODE 1 OR 2
11594  041636  000410                          BR      TST611          ;;GO TO SCOPE EXIT
11595
11596  041640  103003                   4$:     BCC     7$              ;BR IF JMP CLEARED "C"
11597  041642  102002                           BVC     7$              ;BR IF JMP CLEARED "V"
11598  041644  001001                           BNE     7$              ;BR IF JMP CLEARED "Z"
11599  041646  100404                           BMI     TST611          ;;BR IF "N" STILL SET
11600
11601  041650  104006                   7$:     ERROR   6               ;JMP ALTERED CODES - CLEARED
11602  041652  000402                          BR      TST611          ;;GO TO SCOPE EXIT
11603
11604  041654  041640                   8$:     4$                      ;THIS LOCATION CONTAINS JMP ADDRESS
11605
11606  041656  104006                           ERROR   6               ;JMP EXECUTED LIKE MODE 6
11607
11608                                   ;;*****************************************************************
```

```
11609                                   ;*TEST 611      JMP TEST MODE 7; FLAGS = 0000
11610                                   ;;*****************************************************************
11611  041660                           TST611:
11612  041660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11613  041662  012700  000611                  MOV     #611,R0         ;;LOAD R0 WITH TEST NUMBER
11614  041666  013701  041700                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11615  041672  012702  041710                  MOV     #5$,R2  ;[R2] = BASE ADDRESS
11616  041676  000257                          CCC                     ;MAKE N:C = 0000
11617
11618  041700  000172  000020          2$:     JMP     @8$-5$(R2)      ;TEST THE JMP - GO TO 4$
11619
11620  041704  104006                  3$:     ERROR   6               ;JMP FAILED TO LOAD PC
11621  041706  000412                          BR      TST612          ;;GO TO SCOPE EXIT
11622
11623  041710  104006                  5$:     ERROR   6               ;JMP FAILED TO INDEX
11624  041712  000410                          BR      TST612          ;;GO TO SCOPE EXIT
11625
11626  041714  103403                  4$:     BCS     7$              ;BR IF JMP SET "C"
11627  041716  102402                          BVS     7$              ;BR IF JMP SET "V"
11628  041720  001401                          BEQ     7$              ;BR IF JMP SET "Z"
11629  041722  100004                          BPL     TST612          ;;BR IF "N" STILL CLEAR
11630
11631  041724  104006                  7$:     ERROR   6               ;JMP ALTERED CODES - SET
11632  041726  000402                          BR      TST612          ;;GO TO SCOPE EXIT
11633
11634  041730  041714                  8$:     4$                      ;THIS LOCATION CONTAINS JUMP ADDRESS
11635
11636  041732  104006                          ERROR   6               ;JMP EXECUTED LIKE A MODE 6
11637
11638                                   ;;*****************************************************************
11639                                   ;*TEST 612      JSR MODE 1 TEST - LOAD PC  /  PUSH SP
11640                                   ;;*****************************************************************
11641  041734                           TST612:
11642  041734  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11643  041736  012700  000612                  MOV     #612,R0         ;;LOAD R0 WITH TEST NUMBER
11644  041742  013701  041764                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11645  041746  010605                          MOV     SP,R5           ;SAVE THE SP
11646  041750  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11647  041754  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11648  041756  012702  041770                  MOV     #4$,R2          ;DEST ADDR = 4$
11649  041762  000257                          CCC                     ;SCOPE SYNC
11650
11651  041764  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11652
11653  041766  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11654
11655  041770  005726                  4$:     TST     (SP)+           ;POP THE SP
11656  041772  020605                          CMP     SP,R5           ;DID JSR PUSH THE SP ?
11657  041774  001406                          BEQ     TST613          ;;BR IF YES
11658
11659  041776  005746                          TST     -(SP)           ;RESTORE ERROR SP
11660  042000  010603                          MOV     SP,R3           ;[R3]= WAS SP
11661  042002  010504                          MOV     R5,R4
11662  042004  005744                          TST     -(R4)           ;[R4]= S/B SP
11663  042006  104003                  5$.     ERROR   3               ;JSR FAILED TO PUSH THE SP
11664
```

```
11665  042010  010506                          MOV     R5,SP           ;RESTORE SP IN CASE OF ERROR
11666                                  ;;*******************************************************************
11667                                  ;*TEST 613      JSR MODE 1 TEST - CHECK RN AND OLD PC
11668                                  ;;*******************************************************************
11669  042012                          TST613:
11670  042012  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11671  042014  012700  000613                 MOV     #613,R0         ;;LOAD R0 WITH TEST NUMBER
11672  042020  013701  042052                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11673  042024  010605                          MOV     SP,R5           ;SAVE THE SP
11674  042026  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11675  042032  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11676  042034  012702  042056                 MOV     #4$,R2          ;DEST ADDR = 4$
11677  042040  005066  177776                 CLR     -2(SP)          ;INIT STACK LOC TO GET [R4]
11678  042044  012704  125252                 MOV     #125252,R4      ;INIT RN = 125252
11679  042050  000257                          CCC                     ;SCOPE SYNC
11680
11681  042052  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11682
11683  042054  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11684
11685  042056  022726  125252          4$:     CMP     #125252,(SP)+   ;DID JSR SAVE REG ON STACK
11686  042062  001401                          BEQ     8$              ;BR IF IT DID
11687
11688  042064  104005                  5$:     ERROR   5               ;JSR FAILED TO SAVE REG ON STACK
11689
11690  042066  022704  042054          8$:     CMP     #3$,R4          ;DID OLD PC GET SAVED ?
11691  042072  001401                          BEQ     6$              ;BR IF YES
11692
11693  042074  104005                  7$:     ERROR   5               ;JSR FAILED TO SAVE TH OLD PC
11694
11695  042076  010506                  6$:     MOV     R5,SP           ;RESTORE SP IN CASE ERROR SCREWED IT UP
11696
11697                                  ;;*******************************************************************
11698                                  ;*TEST 614      JSR MODE 1 TEST - N:C = 0000
11699                                  ;;*******************************************************************
11700  042100                          TST614:
11701  042100  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11702  042102  012700  000614                 MOV     #614,R0         ;;LOAD R0 WITH TEST NUMBER
11703  042106  013701  042142                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11704                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT12
11705  042112  032737  010000  063234          BIT     #BIT12,@#BPTLOC ;BREAKPOINT HALT SET ??
11706  042120  001401                          BEQ     .+4             ;BR IF NOT
11707  042122  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
11708  042124  010605                          MOV     SP,R5           ;SAVE THE SP
11709  042126  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11710  042132  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11711  042134  012702  042146                 MOV     #4$,R2          ;DEST ADDR = 4$
11712  042140  000257                          CCC                     ;N:C = 0000
11713
11714  042142  004412                  2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11715
11716  042144  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11717
11718  042146  100403                  4$:     BMI     5$              ;N:C = 0000 ?
11719  042150  001402                          BEQ     5$
11720  042152  102401                          BVS     5$
```

```
11721  042154  103001                  BCC     6$
11722
11723  042156  104005          5$:     ERROR   5               ;JSR FAILED - ALTERED FLAGS
11724
11725  042160  010506          6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11726                          ;;*********************************************************
11727                          ;*TEST 615      JSR MODE 1 TEST - N:C = 1111
11728                          ;;*********************************************************
11729  042162                  TST615:
11730  042162  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11731  042164  012700  000615          MOV     #615,R0         ;;LOAD R0 WITH TEST NUMBER
11732  042170  013701  042212          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11733  042174  010605                  MOV     SP,R5           ;SAVE THE SP
11734  042176  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11735  042202  010506          1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
11736  042204  012702  042216          MOV     #4$,R2          ;DEST ADDR = 4$
11737  042210  000277                  SCC                     ;N:C = 1111
11738
11739  042212  004412          2$:     JSR     R4,(R2)         ;TEST THE JSR - GO TO 4$
11740
11741  042214  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11742
11743  042216  100003          4$:     BPL     5$              ;N:C = 1111 ?
11744  042220  001002                  BNE     5$
11745  042222  102001                  BVC     5$
11746  042224  103401                  BCS     6$
11747  042226  104005          5$:     ERROR   5               ;JSR ALTERED FLAGS
11748
11749  042230  010506          6$:     MOV     R5,SP           ;RESET SP IN CASE OF ERROR
11750
11751                          ;;*********************************************************
11752                          ;*TEST 616      JSR MODE 2 TEST
11753                          ;;*********************************************************
11754  042232                  TST616:
11755  042232  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11756  042234  012700  000616          MOV     #616,R0         ;;LOAD R0 WITH TEST NUMBER
11757  042240  013701  042262          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11758  042244  010605                  MOV     SP,R5           ;SAVE THE SP
11759  042246  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11760  042252  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11761  042254  012702  042266          MOV     #4$,R2          ;DEST ADDR = 4$
11762  042260  000257                  CCC                     ;SCOPE SYNC
11763
11764  042262  004422          2$:     JSR     R4,(R2)+        ;TEST THE JSR - GO TO 4$
11765
11766  042264  104006          3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11767
11768  042266  005726          4$:     TST     (SP)+           ;RESET SP
11769  042270  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11770  042272  001406                  BEQ     TST617          ;;BR IF YES
11771
11772  042274  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11773  042276  010603                  MOV     SP,R3           ;WAS SP
11774  042300  010504                  MOV     R5,R4
11775  042302  005744                  TST     -(R4)           ;S/B SP
11776  042304  104003          5$:     ERROR   3               ;JSR FAILED TO PUSH SP
```

```
11777
11778  042306  010506                  MOV     R5,SP           ;RESTORE SP JUST IN CASE
11779
11780                                  ;;*****************************************************************
11781                                  ;*TEST 617      JSR MODE 3 TEST
11782                                  ;;*****************************************************************
11783  042310                          TST617:
11784  042310  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11785  042312  012700  000617                 MOV     #617,R0         ;;LOAD R0 WITH TEST NUMBER
11786  042316  013701  042340                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11787  042322  010605                         MOV     SP,R5           ;SAVE THE SP
11788  042324  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11789  042330  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11790  042332  012702  042366                 MOV     #7$,R2          ;DEST ADDR = [7$]
11791  042336  000257                         CCC                     ;SCOPE SYNC
11792
11793  042340  004432                  2$:    JSR     R4,@(R2)+       ;TEST THE JSR - GO TO 4$ VIA 7$
11794
11795  042342  104006                  3$:    ERROR   6               ;JSR FAILED TO LOAD THE PC
11796
11797  042344  005726                  4$:    TST     (SP)+           ;RESET SP
11798  042346  020605                         CMP     SP,R5           ;DID JSR PUSH STACK ?
11799  042350  001411                         BEQ     TST620          ;;BR IF YES
11800
11801  042352  005746                         TST     -(SP)           ;RESET SP TO ERROR VALUE
11802  042354  010603                         MOV     SP,R3           ;WAS SP
11803  042356  010504                         MOV     R5,R4
11804  042360  005744                         TST     -(R4)           ;S/B SP
11805  042362  104003                  5$:    ERROR   3               ;JSR FAILED
11806  042364  000402                         BR      6$              ;GO EXIT
11807
11808  042366  042344                  7$:    4$                      ;CONTAINS JUMP ADDR
11809  042370  104006                         ERROR   6               ;JSR EXECUTED LIKE A MODE 1 OR 2
11810
11811  042372  010506                  6$:    MOV     R5,SP           ;RESTORE SP JUST IN CASE
11812
11813                                  ;;*****************************************************************
11814                                  ;*TEST 620      JSR MODE 4 TEST
11815                                  ;;*****************************************************************
11816  042374                          TST620:
11817  042374  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11818  042376  012700  000620                 MOV     #620,R0         ;;LOAD R0 WITH TEST NUMBER
11819  042402  013701  042424                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11820  042406  010605                         MOV     SP,R5           ;SAVE THE SP
11821  042410  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11822  042414  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11823  042416  012702  042432                 MOV     #5$,R2          :DEST ADDR = 4$+2
11824  042422  000257                         CCC                     ;SCOPE SYNC
11825
11826  042424  004442                  2$:    JSR     R4,-(R2)        ;TEST THE JSR - GO TO 4$
11827
11828  042426  104006                  3$:    ERROR   6               ;JSR FAILED TO LOAD THE PC
11829
11830  042430  000401                  4$:    BR      6$              ;JUMPED OK - GO CHECK SP
11831  042432  104005                  5$:    ERROR   5               ;JSR FAILED TO DECREMENT DEST REG
11832
```

```
11833  042434  005726                  6$:     TST     (SP)+           ;RESET SP
11834  042436  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11835  042440  001406                          BEQ     TST621          ;;BR IF YES
11836
11837  042442  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11838  042444  010603                          MOV     SP,R3           ;WAS SP
11839  042446  010504                          MOV     R5,R4
11840  042450  005744                          TST     -(R4)           ;S/B SP
11841  042452  104003                  7$:     ERROR   3               ;JSR FAILED TO PUSH SP
11842
11843  042454  010506                  8$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11844
11845                                  ;;*****************************************************************
11846                                  ;*TEST 621       JSR MODE 5 TEST
11847                                  ;;*****************************************************************
11848  042456                          TST621:
11849  042456  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11850  042460  012700  000621                  MOV     #621,R0         ;;LOAD R0 WITH TEST NUMBER
11851  042464  013701  042506                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11852  042470  010605                          MOV     SP,R5           ;SAVE THE SP
11853  042472  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11854  042476  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11855  042500  012702  042536                  MOV     #7$,R2          ;DEST ADDR = [7$ - 2]
11856  042504  000257                          CCC                     ;SCOPE SYNC
11857
11858  042506  004452                  2$:     JSR     R4,@-(R2)       ;TEST THE JSR - GO TO 4$
11859
11860  042510  104006                  3$:     ERROR   6               ;JSR FAILED TO LOAD THE PC
11861
11862  042512  005726                  4$:     TST     (SP)+           ;RESET SP
11863  042514  020605                          CMP     SP,R5           ;DID JSR PUSH STACK ?
11864  042516  001411                          BEQ     TST622          ;;BR IF YES
11865
11866  042520  005746                          TST     -(SP)           ;RESET SP TO ERROR VALUE
11867  042522  010603                          MOV     SP,R3           ;WAS SP
11868  042524  010504                          MOV     R5,R4
11869  042526  005744                          TST     -(R4)           ;S/B SP
11870  042530  104003                  5$:     ERROR   3               ;JSR FAILED TO PUSH SP
11871  042532  000402                          BR      6$              ;GO EXIT
11872
11873  042534  042512                          4$                      ;CONTAINS JUMP ADDRESS
11874  042536  104005                  7$:     ERROR   5               ;JSR EXECUTED LIKE A MODE 1 OR 2
11875
11876  042540  010506                  6$:     MOV     R5,SP           ;RESTORE SP JUST IN CASE
11877
11878                                  ;;*****************************************************************
11879                                  ;*TEST 622       JSR MODE 6 TEST
11880                                  ;;*****************************************************************
11881  042542                          TST622:
11882  042542  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11883  042544  012700  000622                  MOV     #622,R0         ;;LOAD R0 WITH TEST NUMBER
11884  042550  013701  042572                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11885  042554  010605                          MOV     SP,R5           ;SAVE THE SP
11886  042556  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11887  042562  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11888  042564  012702  042576                  MOV     #3$,R2          ;[R2] = BASE DEST ADDR
```

```
11889  042570  000257                  CCC                     ;SCOPE SYNC
11890
11891  042572  004462  000002    2$:   JSR     R4,4$-3$(R2)    ;TEST THE JSR - GO TO 4$
11892
11893  042576  104006              3$: ERROR   6               ;JSR FAILED TO LOAD THE PC OR INDEX FAILED
11894
11895  042600  005726              4$: TST     (SP)+           ;RESET SP
11896  042602  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11897  042604  001406                  BEQ     TST623          ;;BR IF YES
11898
11899  042606  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11900  042610  010603                  MOV     SP,R3           ;WAS SP
11901  042612  010504                  MOV     R5,R4
11902  042614  005744                  TST     -(R4)           ;S/B SP
11903  042616  104003              5$: ERROR   3               ;JSR FAILED TO PUSH STACK
11904  042620  010506                  MOV     R5,SP           ;RESET SP JUST IN CASE
11905
11906                          ;;*****************************************************************
11907                          ;*TEST 623      JSR MODE 7 TEST
11908                          ;;*****************************************************************
11909  042622                  TST623:
11910  042622  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11911  042624  012700  000623          MOV     #623,R0         ;;LOAD R0 WITH TEST NUMBER
11912  042630  013701  042652          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11913  042634  010605                  MOV     SP,R5           ;SAVE THE SP
11914  042636  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
11915  042642  010506              1$: MOV     R5,SP           ;RESET SP FOR ERROR LOOPS
11916  042644  012702  042656          MOV     #3$,R2          ;BASE DEST ADDR = 3$
11917  042650  000257                  CCC                     ;SCOPE SYNC
11918
11919  042652  004472  000024    2$:   JSR     R4,@7$-3$(R2)   ;TEST THE JSR - GO TO 4$ VIA 7$
11920
11921  042656  104006              3$: ERROR   6               ;JSR FAILED TO LOAD THE PC
11922                                                          ;OR THE INDEX FAILED
11923
11924  042660  005726              4$: TST     (SP)+           ;RESET SP
11925  042662  020605                  CMP     SP,R5           ;DID JSR PUSH STACK ?
11926  042664  001411                  BEQ     TST624          ;;BR IF YES
11927
11928  042666  005746                  TST     -(SP)           ;RESET SP TO ERROR VALUE
11929  042670  010603                  MOV     SP,R3           ;WAS SP
11930  042672  010504                  MOV     R5,R4
11931  042674  005744                  TST     -(R4)           ;S/B SP
11932  042676  104003              5$: ERROR   3               ;JSR FAILED TO PUSH STACK
11933  042700  000402                  BR      6$              ;SKIP TO EXIT
11934
11935  042702  042660              7$: 4$                      ;CONTAINS JUMP ADDR
11936  042704  104005                  ERROR   5               ;JSR WORKED LIKE A MODE 1 OR 2
11937
11938  042706  010506              6$: MOV     R5,SP           ;RESTORE SP JUST IN CASE
11939
11940                          ;;*****************************************************************
11941                          ;*TEST 624      SOB TEST, [R] = 1, NO BRANCH
11942                          ;;*****************************************************************
11943  042710                  TST624:
11944  042710  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
11945  042712  012700  000624          MOV     #624,R0         ;;LOAD R0 WITH TEST NUMBER
11946  042716  013701  042736          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11947  042722  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
11948  042726  000402                  BR      2$-2            ;GO DO THE SOB
11949
11950  042730  104006          3$:     ERROR   6               ;SOB SHOULDN'T HAVE BRANCHED HERE
11951  042732  000402                  BR      TST625          ;;GO TO SCOPE CALL
11952
11953  042734  000257                  CCC                     ;SYNC INSTR.
11954  042736  077204          2$:     SOB     R2,3$           ;TEST THE SOB
11955
11956                          ;;*********************************************************************
11957                          ;*TEST 625       SOB TEST, [R] = 5, BRANCH 4 TIMES
11958                          ;;*********************************************************************
11959  042740                  TST625:
11960  042740  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11961  042742  012700  000625          MOV     #625,R0         ;;LOAD R0 WITH TEST NUMBER
11962  042746  013701  043000          MOV     @#SOB2,R1       ;GET COPY OF TEST INSTRUCTION WORD
11963  042752  012702  000005          MOV     #5,R2           ;SET SOB COUNTER = 5
11964  042756  012705  177773          MOV     #-5,R5          ;SET UP R5 TO COUNT 5 BRANCHES
11965  042762  000405                  BR      SOB2-2          ;GO DO THE SOB
11966
11967  042764  000474          SOB1:   BR      SOB3            ;USED BY LAST SOB TEST TO TEST MAX OFFSET
11968  042766  000240                  NOP                     ;OFFSET ADJUSTMENT
11969  042770  000240                  NOP
11970
11971  042772  005205          SOB5:   INC     R5              ;COUNT ONE BRANCH
11972  042774  001406                  BEQ     SOBERR          ;BR IF TOO MANY LOOPS BY SOB
11973
11974  042776  000257                  CCC                     ;SCOPE SYNC
11975  043000  077204          SOB2:   SOB     R2,SOB5         ;TEST THE SOB
11976  043002  005702                  TST     R2              ;R2 SHOULD CONTAIN 0
11977  043004  001403                  BEQ     TST626          ;;BR IF IT DOES
11978
11979  043006  104006                  ERROR   6               ;SOB COUNTER NOT ZERO
11980  043010  000401                  BR      TST626          ;;GO TO SCOPE CALL
11981  043012  104006          SOBERR: ERROR   6               ;SOB MADE TOO MANY BRANCHES
11982
11983                          ;;*********************************************************************
11984                          ;*TEST 626       SOB TEST, [R] = 1, FLAGS = 1111
11985                          ;;*********************************************************************
11986  043014                  TST626:
11987  043014  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
11988  043016  012700  000626          MOV     #626,R0         ;;LOAD R0 WITH TEST NUMBER
11989  043022  013701  043034          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
11990  043026  012702  000001          MOV     #1,R2           ;SET SOB COUNTER = 1
11991  043032  000277                  SCC                     ;MAKE N:C = 1111
11992
11993  043034  077202          2$:     SOB     R2,2$-2         ;TEST THE SOB
11994
11995  043036  103003                  BCC     3$              ;BR IF C = 0
11996  043040  102002                  BVC     3$              ;BR IF V = 0
11997  043042  001001                  BNE     3$              ;BR IF Z = 0
11998  043044  100401                  BMI     TST627          ;;BR IF N = 1
11999
12000  043046  104006          3$:     ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
```

```
12001
12002                                   ;;*****************************************************************
12003                                   ;*TEST 627       SOB TEST, [R] = 1, FLAGS = 0000
12004                                   ;;*****************************************************************
12005  043050                           TST627:
12006  043050  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12007  043052  012700  000627                  MOV     #627,R0         ;;LOAD R0 WITH TEST NUMBER
12008  043056  013701  043070                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12009  043062  012702  000001                  MOV     #1,R2           ;SET SOB COUNTER = 1
12010  043066  000257                          CCC                     ;MAKE N:C = 0000
12011
12012  043070  077202                   2$:    SOB     R2,2$-2         ;TEST THE SOB
12013
12014  043072  103403                          BCS     3$              ;BR IF C = 1
12015  043074  102402                          BVS     3$              ;BR IF V = 1
12016  043076  001401                          BEQ     3$              ;BR IF Z = 1
12017  043100  100001                          BPL     TST630          ;;BR IF N = 0
12018
12019  043102  104006                   3$:    ERROR   6               ;SOB ALTERED CODES - SET ONE
12020
12021                                   ;;*****************************************************************
12022                                   ;*TEST 630       SOB TEST, [R] = 5, FLAGS = 1111
12023                                   ;;*****************************************************************
12024  043104                           TST630:
12025  043104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12026  043106  012700  000630                  MOV     #630,R0         ;;LOAD R0 WITH TEST NUMBER
12027  043112  013701  043124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12028  043116  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12029  043122  000277                          SCC                     ;MAKE N:C = 1111
12030
12031  043124  077201                   2$:    SOB     R2,2$           ;TEST THE SOB
12032
12033  043126  103003                          BCC     3$              ;BR IF C = 0
12034  043130  102002                          BVC     3$              ;BR IF V = 0
12035  043132  001001                          BNE     3$              ;BR IF Z = 0
12036  043134  100401                          BMI     TST631          ;;BR IF N = 1
12037
12038  043136  104006                   3$:    ERROR   6               ;SOB ALTERED CODES - CLEARED ONE
12039
12040                                   ;;*****************************************************************
12041                                   ;*TEST 631       SOB TEST, [R] = 5, FLAGS = 0000
12042                                   ;;*****************************************************************
12043  043140                           TST631:
12044  043140  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12045  043142  012700  000631                  MOV     #631,R0         ;;LOAD R0 WITH TEST NUMBER
12046  043146  013701  043160                  MOV     @#SOB4,R1       ;GET COPY OF TEST INSTRUCTION WORD
12047  043152  012702  000005                  MOV     #5,R2           ;SET SOB COUNTER = 5
12048  043156  000257                   SOB3:  CCC                     ;MAKE N:C = 0000
12049
12050  043160  077277                   SOB4:  SOB     R2,SOB1         ;TEST THE SOB
12051
12052  043162  103403                          BCS     3$              ;BR IF C = 1
12053  043164  102402                          BVS     3$              ;BR IF V = 1
12054  043166  001401                          BEQ     3$              ;BR IF Z = 1
12055  043170  100001                          BPL     TST632          ;;BR IF N = 0
12056
```

```
12057  043172  104006                  3$:     ERROR   6               ;SOB ALTERED CODES - SET ONE
12058
12059                                  ;;*****************************************************************
12060                                  ;*TEST 632       RTS TEST - N:C = 0000
12061                                  ;;*****************************************************************
12062  043174                          TST632:
12063  043174  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12064  043176  012700  000632                 MOV     #632,R0         ;;LOAD R0 WITH TEST NUMBER
12065  043202  013701  043234                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12066  043206  010605                         MOV     SP,R5           ;SAVE THE SP
12067  043210  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12068  043214  012704  177777          1$:    MOV     #-1,R4          ;R3 SHOULD GET 177777
12069  043220  010506                         MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12070  043222  012703  043242                 MOV     #4$,R3          ;RTS SHOULD LOAD PC FROM [R3]
12071  043226  012746  177777                 MOV     #-1,-(SP)       ;RTS SHOULD LOAD R3 WITH 177777
12072  043232  000257                         CCC                     ;N:C = 0000
12073
12074  043234  000203                  2$:    RTS     R3              ;TEST THE RTS - GO TO 4$
12075
12076  043236  104005                  3$:    ERROR   5               ;RTS FAILED TO LOAD THE PC
12077  043240  000415                         BR      10$             ;GO TO EXIT - SCHOOLS OUT
12078
12079  043242  100403                  4$:    BMI     5$              ;N:C = 0000 ?
12080  043244  001402                         BEQ     5$
12081  043246  102401                         BVS     5$
12082  043250  103001                         BCC     6$
12083
12084  043252  104005                  5$:    ERROR   5               ;RTS ALTERED CODES - CLEARED ONE
12085
12086  043254  020403                  6$:    CMP     R4,R3           ;DID R3 GET LOADED FROM STACK ?
12087  043256  001401                         BEQ     8$              ;BR IF YES
12088
12089  043260  104002                  7$:    ERROR   2               ;RTS FAILED TO LOAD REG
12090
12091  043262  020506                  8$:    CMP     R5,SP           ;DID RTS POP THE STACK POINTER ?
12092  043264  001404                         BEQ     TST633          ;;BR IF YES
12093
12094  043266  010504                         MOV     R5,R4           ;[R4] = S / B SP
12095  043270  010603                         MOV     SP,R3           ;[R3] = WAS SP
12096  043272  104003                  9$:    ERROR   3               ;RTS FAILED TO POP SP
12097
12098  043274  010506                  10$:   MOV     R5,SP           ;FIX THE SP
12099
12100                                  ;;*****************************************************************
12101                                  ;*TEST 633       RTT TEST - N:C = 1111
12102                                  ;;*****************************************************************
12103  043276                          TST633:
12104  043276  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12105  043300  012700  000633                 MOV     #633,R0         ;;LOAD R0 WITH TEST NUMBER
12106  043304  013701  043346                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12107  043310  012702  177776                 MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12108  043314  010605                         MOV     SP,R5           ;SAVE THE SP
12109  043316  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12110  043322  010506                  1$:    MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12111  043324  012704  000340                 MOV     #340,R4         ;[R4] = S / B PSW AT HTIS POINT
12112  043330  012746  000340                 MOV     #340,-(SP)      ;NEW PSW S / B - 340
```

```
12113  043334  012746  043354          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12114  043340  005037  177776          CLR     @#PSW           ;CLEAR THE PSW
12115  043344  000277                  SCC                     ;N:C = 1111
12116
12117  043346  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12118
12119  043350  104005          3$:     ERROR   5               ;RTT FAILED TO LOAD THE PC
12120  043352  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12121
12122  043354  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12123  043360  020403                  CMP     R4,R3           ;WAS PSW = 340 ?
12124  043362  001401                  BEQ     6$              ;BR IF IT WAS
12125
12126  043364  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12127
12128  043366  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12129  043370  001404                  BEQ     TST634          ;;BR IF YES
12130
12131  043372  010504                  MOV     R5,R4           ;[R4] = S / B SP
12132  043374  010603                  MOV     SP,R3           ;[R3] = WAS SP
12133  043376  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12134
12135  043400  010506          8$:     MOV     R5,SP           ;FIX THE SP
12136
12137                          ;;*******************************************************
12138                          ;*TEST 634       RTT TEST - N:C = 0000
12139                          ;;*******************************************************
12140  043402                  TST634:
12141  043402  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12142  043404  012700  000634          MOV     #634,R0         ;;LOAD R0 WITH TEST NUMBER
12143  043410  013701  043454          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12144  043414  012702  177776          MOV     #PSW,R2         ;DEST=PSW FOR 5$ CALL
12145  043420  010605                  MOV     SP,R5           ;SAVE THE SP
12146  043422  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12147  043426  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12148  043430  012704  000017          MOV     #017,R4         ;[R4] = S / B PSW AT HTIS POINT
12149  043434  012746  000017          MOV     #017,-(SP)      ;NEW PSW S / B = 017
12150  043440  012746  043462          MOV     #4$,-(SP)       ;NEW PC S / B = 4$
12151  043444  012737  000340  177776  MOV     #340,@#PSW      ;MAKE [PSW] = 340
12152  043452  000257                  CCC                     ;N:C = 0000
12153
12154  043454  000006          2$:     RTT                     ;TEST THE RTT - GO TO 4$
12155
12156  043456  104005          3$:     ERROR   5               ;RTT FAILED TC LOAD THE PC
12157  043460  000412                  BR      8$              ;GO TO EXIT - SCHOOL'S OUT
12158
12159  043462  013703  177776  4$:     MOV     @#PSW,R3        ;SAVE THE PSW
12160  043466  020403                  CMP     R4,R3           ;WAS PSW = 017 ?
12161  043470  001401                  BEQ     6$              ;BR IF IT WAS
12162
12163  043472  104001          5$:     ERROR   1               ;RTT FAILED TO LOAD PSW PROPERLY
12164
12165  043474  020506          6$:     CMP     R5,SP           ;DID RTT UPDATE THE SP ?
12166  043476  001404                  BEQ     TST635          ;;BR IF YES
12167
12168  043500  010504                  MOV     R5,R4           ;[R4] - S / B SP
```

```
12169  043502  010603                  MOV     SP,R3           ;[R3] = WAS SP
12170  043504  104003          7$:     ERROR   3               ;RTT FAILED TO UPDATE SP
12171
12172  043506  010506          8$:     MOV     R5,SP           ;FIX THE SP
12173
12174                          ;;**********************************************************
12175                          ;*TEST 635       MARK INSTRUCTION TEST - N:C=0000
12176                          ;;**********************************************************
12177  043510                  TST635:
12178  043510  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12179  043512  012700  000635          MOV     #635,R0         ;;LOAD R0 WITH TEST NUMBER
12180  043516  013701  043542          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12181  043522  010602                  MOV     SP,R2           ;SAVE SP
12182  043524  012704  125252          MOV     #125252,R4      ;[R5] SHOULD BE 125252
12183  043530  012705  043572          MOV     #4$,R5          ;MARK GOES TO 4$ VIA [R5]
12184  043534  010437  043556          MOV     R4,@#6$         ;INITIALIZE WORD LOADED INTO R5
12185  043540  000257                  CCC                     ;N:C=0000
12186
12187  043542  006405          2$:     MARK+5                  ;TEST THE MARK
12188
12189  043544  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12190  043550  010206                  MOV     R2,SP           ;RESET SP
12191  043552  104005          3$:     ERROR   5               ;MARK FAILED TO EXECUTE
12192
12193  043554  000444                  BR      TST636          ;;GO TO SCOPE EXIT
12194
12195  043556  125252          6$:     125252                  ;THIS WORD SHOULD GET LOADED INTO R5
12196
12197  043560  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12198  043564  010206                  MOV     R2,SP           ;RESET SP
12199  043566  104005          5$:     ERROR   5               ;MARK FAILED TO LOAD RC FROM [R5]
12200
12201  043570  000436                  BR      TST636          ;;GO TO SCOPE EXIT
12202
12203  043572  100403          4$:     BMI     10$             ;N:C-0000?
12204  043574  001402                  BEQ     10$
12205  043576  102401                  BVS     10$
12206  043600  103011                  BCC     8$
12207
12208  043602  013703  177776  10$:    MOV     @#PSW,R3        ;SAVE FLAGS IN R3
12209  043606  010637  001074          MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12210  043612  010206                  MOV     R2,SP           ;RESET SP
12211  043614  012702  177776          MOV     #PSW,R2         ;DEST=PSW
12212  043620  104007          7$:     ERROR   7               ;MARK SET A FLAG
12213  043622  000421                  BR      TST636          ;;GO TO SCOPE EXIT
12214
12215  043624  020627  043560  8$:     CMP     SP,#6$+2        ;DID MARK RESET SP?
12216  043630  001406                  BEQ     11$             ;BR IF YES
12217  043632  010603                  MOV     SP,R3           ;PUT BAD SP IN R3
12218  043634  012704  043560          MOV     #6$+2,R4        ;S/B SP
12219  043640  010206                  MOV     R2,SP           ;RESET SP
12220  043642  104003          9$:     ERROR   3               ;MARK FAILED TO RESET SP
12221
12222  043644  000410                  BR      TST636          ;;GO TO SCOPE EXIT
12223
12224  043646  020504          11$:    CMP     R5,R4           ;DID MARK RESTORE OLD R5
```

```
 12225  043650  001405                          BEQ     12$                 ;BR IF YES
 12226
 12227  043652  010637  001074                  MOV     SP,@#$REG5          ;SAVE BAD SP FOR PRINTING
 12228  043656  010503                          MOV     R5,R3               ;WAS DEST
 12229  043660  010206                          MOV     R2,SP               ;RESET SP
 12230  043662  104004                          ERROR   4                   ;MARK FAILED TO RESET R5
 12231
 12232  043664  010206                  12$:    MOV     R2,SP               ;RESET SP
 12233
 12234                                  ;;**********************************************************************
 12235                                  ;*TEST 636      MARK INSTRUCTION TEST - N:C=1111
 12236                                  ;;**********************************************************************
 12237  043666                          TST636:
 12238  043666  000004                          SCOPE                       ;CALL THE SCOPE LOOP UTILITY
 12239  043670  012700  000636                  MOV     #636,R0             ;;LOAD R0 WITH TEST NUMBER
 12240  043674  013701  043720                  MOV     @#2$,R1             ;LOAD R1 WITH TEST INSTRUCTION WORD
 12241  043700  010602                          MOV     SP,R2               ;SAVE SP
 12242  043702  012704  125252                  MOV     #125252,R4          ;[R5] SHOULD BE 125252
 12243  043706  012705  043750                  MOV     #4$,R5              ;MARK GOES TO 4$ VIA [R5]
 12244  043712  010437  043734                  MOV     R4,@#6$             ;INITIALIZE WORD LOADED INTO R5
 12245  043716  000277                          SCC                         ;N:C=1111
 12246
 12247  043720  006405                  2$:     MARK+5                      ;TEST THE MARK
 12248
 12249  043722  010637  001074                  MOV     SP,@#$REG5          ;SAVE BAD SP FOR PRINTING
 12250  043726  010206                          MOV     R2,SP               ;RESET SP
 12251  043730  104005                  3$:     ERROR   5                   ;MARK FAILED TO EXECUTE
 12252
 12253  043732  000444                          BR      TST637              ;;GO TO SCOPE EXIT
 12254
 12255  043734  125252                  6$:     125252                      ;THIS WORD SHOULD GET LOADED INTO R5
 12256
 12257  043736  010637  001074                  MOV     SP,@#$REG5          ;SAVE BAD SP FOR PRINTING
 12258  043742  010206                          MOV     R2,SP               ;RESET SP
 12259  043744  104005                  5$:     ERROR   5                   ;MARK FAILED TO LOAD RC FROM [R5]
 12260
 12261  043746  000436                          BR      TST637              ;;GO TO SCOPE EXIT
 12262
 12263  043750  100003                  4$:     BPL     7$                  ;N:C=1111
 12264  043752  001002                          BNE     7$
 12265  043754  102001                          BVC     7$
 12266  043756  103411                          BCS     8$
 12267
 12268  043760  013703  177776          7$:     MOV     @#PSW,R3            ;SAVE FLAGS IN R3
 12269  043764  010637  001074                  MOV     SP,@#$REG5          ;SAVE BAD SP FOR PRINTING
 12270  043770  010206                          MOV     R2,SP               ;RESET SP
 12271  043772  012702  177776                  MOV     #PSW,R2             ;DEST=PSW
 12272  043776  104007                          ERROR   7                   ;MARK SET A FLAG
 12273  044000  000421                          BR      TST637              ;;GO TO SCOPE EXIT
 12274
 12275  044002  020627  043736          8$:     CMP     SP,#6$+2            ;DID MARK RESET SP?
 12276  044006  001406                          BEQ     9$                  ;BR IF YES
 12277  044010  010603                          MOV     SP,R3               ;PUT BAD SP IN R3
 12278  044012  012704  043736                  MOV     #6$+2,R4            ;S/B SP
 12279  044016  010206                          MOV     R2,SP               ;RESET SP
 12280  044020  104003                          ERROR   3                   ;MARK FAILED TO RESET SP
```

```
12281
12282  044022  000410                          BR      TST637          ;;GO TO SCOPE EXIT
12283
12284  044024  020504                  9$:     CMP     R5,R4           ;DID MARK RESTORE OLD R5
12285  044026  001405                          BEQ     10$             ;BR IF YES
12286
12287  044030  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12288  044034  010503                          MOV     R5,R3           ;WAS DEST
12289  044036  010206                          MOV     R2,SP           ;RESET SP
12290  044040  104004                          ERROR   4               ;MARK FAILED TO RESET R5
12291
12292  044042  010206                  10$:    MOV     R2,SP           ;RESET SP
12293
12294                                  ;;**********************************************************
12295                                  ;*TEST 637       BASIC LINE CLOCK RESPONSE TEST
12296                                  ;;**********************************************************
12297  044044                          TST637:
12298  044044  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12299  044046  012700  000637                  MOV     #637,R0         ;;LOAD R0 WITH TEST NUMBER
12300  044052  013701  044102                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12301  044056  010605                          MOV     SP,R5           ;SAVE SP
12302  044060  012702  177546                  MOV     #LKCSR,R2       ;[R2] = LINE CLOCK ADDRESS
12303  044064  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12304  044070  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12305  044072  012737  044106  000004          MOV     #4$,@#4         ;GO TO 4$ IF BUS TIMEOUT
12306  044100  000257                          CCC                     ;SCOPE SYNC
12307
12308  044102  005712                  2$:     TST     (R2)            ;REFERENCE LKCSR ADDR
12309
12310  044104  000404                          BR      6$              ;GO TO EXIT
12311
12312  044106  012737  061220  000004  4$:     MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12313  044114  104006                  3$:     ERROR   6               ;LKCSR FAILED TO RESPOND
12314
12315  044116  010506                  6$:     MOV     R5,SP           ;RESET SP
12316  044120  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE TIMEOUT VECTOR
12317
12318                                  ;;**********************************************************
12319                                  ;*TEST 640       LINE CLOCK TEST - LKCSR BIT 7 SET
12320                                  ;;**********************************************************
12321  044126                          TST640:
12322  044126  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12323  044130  012700  000640                  MOV     #640,R0         ;;LOAD R0 WITH TEST NUMBER
12324  044134  013701  044152                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12325  044140  012702  177546                  MOV     #LKCSR,R2       ;DEST ADDR = 177546
12326  044144  012704  000200                  MOV     #200,R4         ;[LKCSR] S / B = 200
12327  044150  000257                          CCC                     ;SCOPE SYNC
12328
12329  044152  030412                  2$:     BIT     R4,(R2)         ;TEST BIT 7 IN LKCSR
12330
12331  044154  001002                          BNE     TST641          ;;BR IF IT'S SET
12332
12333  044156  011203                          MOV     (R2),R3         ;GET WAS DATA
12334  044160  104001                  3$:     ERROR   1               ;BIT 7 NOT SET IN LKCSR
12335
12336                                  ;;**********************************************************
```

```
12337                                   ;*TEST 641      LINE CLOCK TEST - LKCSR BIT 6 CLEAR
12338                                   ;;**************************************************************
12339  044162                           TST641:
12340  044162  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12341  044164  012700  000641                  MOV     #641,R0         ;;LOAD R0 WITH TEST NUMBER
12342  044170  013701  044206                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12343  044174  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12344  044200  012704  000200                  MOV     #200,R4         ;[LKCSR] S / B = 200
12345  044204  000257                          CCC                     ;SCOPE SYNC
12346
12347  044206  032712  000100          2$:     BIT     #100,(R2)       ;TEST BIT 6 IN LKCSR
12348
12349  044212  001402                          BEQ     TST642          ;;BR IF CLEAR
12350
12351  044214  011203                          MOV     (R2),R3         ;GET WAS DATA
12352  044216  104001                  3$:     ERROR   1               ;BIT 6 (INTR. ENAB.) IN LKCSR WAS SET
12353
12354                                   ;;**************************************************************
12355                                   ;*TEST 642      LINE CLOCK TEST - LKCSR BIT 6 SET
12356                                   ;;**************************************************************
12357  044220                           TST642:
12358  044220  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12359  044222  012700  000642                  MOV     #642,R0         ;;LOAD R0 WITH TEST NUMBER
12360  044226  013701  044276                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12361  044232  010605                          MOV     SP,R5           ;SAVE SP
12362  044234  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
12363  044240  012704  000300                  MOV     #300,R4         ;[LKCSR] S / B = 300
12364  044244  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12365  044250  012737  044312  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR IN CASE LOGIC
12366  044256  012737  000340  000102          MOV     #340,@#102      ;FAULT CAUSES ATL INTERRUPT
12367  044264  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12368  044266  012737  000340  177776          MOV     #340,@#PSW      ;SET PRIORITY TO LEVEL 7
12369  044274  000257                          CCC                     ;SCOPE SYNC
12370
12371  044276  052712  000100          2$:     BIS     #100,(R2)       ;SET BIT 6 IN LKCSR
12372
12373  044302  020412                          CMP     R4,(R2)         ;RESULT CORRECT?
12374  044304  001402                          BEQ     4$              ;BR IF YES
12375
12376  044306  011203                          MOV     (R2),R3         ;GET WAS DATA
12377  044310  104001                  3$:     ERROR   1               ;BIT 6 FAILED TO SET IN LKCSR
12378
12379  044312  042737  000102  000100  4$:     BIC     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLOCK VECTOR
12380  044320  005037  000102                  CLR     @#102
12381  044324  042712  000100                  BIC     #100,(R2)       ;TURN OF LINE CLK INTR. ENAB.
12382  044330  010506                          MOV     R5,SP           ;RESET SP
12383
12384                                   ;;**************************************************************
12385                                   ;*TEST 643      LINE CLK BASIC INTERRUPT TEST
12386                                   ;;**************************************************************
12387  044332                           TST643:
12388  044332  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12389  044334  012700  000643                  MOV     #643,R0         ;;LOAD R0 WITH TEST NUMBER
12390  044340  013701  044406                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12391  044344  010605                          MOV     SP,R5           ;SAVE SP
12392  044346  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LKCSR
```

```
12393  044352  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12394  044356  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12395  044360  005004                  CLR     R4              ;INITIALIZE TIMER
12396  044362  012737  044424  000100  MOV     #4$,@#100       ;SET UP LINE CLOCK VECTOR TO TO
12397  044370  012737  000340  000102  MOV     #340,@#102      ;TO 4$ WITH PROCESSOR PRIORITY = 7
12398  044376  005012                  CLR     (R2)            ;CLEAR LKCSR
12399  044400  005037  177776          CLR     @#PSW           ;SET PRIORITY TO LEVEL 000
12400  044404  000257                  CCC                     ;SCOPE SYNC
12401
12402  044406  052712  000100  2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPT
12403
12404  044412  005304                  DEC     R4              ;WAIT FOR INTR - REPORT ERROR IF
12405  044414  001376                  BNE     .-2             ;R4 GOES TO 000000
12406
12407  044416  042712  000100          BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12408  044422  104006          3$:     ERROR   6               ;LINE CLK FAILED TO INTERRUPT
12409
12410  044424  042712  000100  4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENAB.
12411  044430  012737  000102  000100  MOV     #102,@#100      ;RESTORE TRAP CATCHER IN LINE CLK VECTOR
12412  044436  005037  000102          CLR     @#102
12413  044442  010506                  MOV     R5,SP           ;RESET SP
12414  044444  005037  177776          CLR     @#PSW           ;RESET PRIORITY TO LEVEL 0
12415
12416                          ;;**********************************************************************
12417                          ;*TEST 644      RESET TEST - N:C = 1111
12418                          ;;**********************************************************************
12419  044450                  TST644:
12420  044450  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12421  044452  012700  000644          MOV     #644,R0         ;;LOAD R0 WITH TEST NUMBER
12422  044456  013701  044510          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12423  044462  012737  000001  001110  MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
12424  044470  012702  177564          MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12425  044474  012737  000340  177776  MOV     #340,@#PSW      ;MAKE PRTY. BITS ALL 1'S
12426  044502  052712  000004          BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12427  044506  000277                  SCC                     ;N:C = 1111
12428
12429  044510  000005          2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12430
12431  044512  013705  177776          MOV     @#PSW,R5        ;SAVE THE PSW
12432  044516  032712  000004          BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12433  044522  001403                  BEQ     4$              ;BR IF YES
12434
12435  044524  042712  000004          BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12436  044530  104006          3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12437
12438  044532  022705  000357  4$:     CMP     #357,R5         ;DID RESET ALTER THE PSW ??
12439  044536  001406                  BEQ     6$              ;BR IF NOT
12440
12441  044540  012704  000357          MOV     #357,R4         ;[R4] = S/B PSW
12442  044544  010503                  MOV     R5,R3           ;[R3] = WAS PSW
12443  044546  012702  177776          MOV     #PSW,R2         ;DEST = PSW
12444  044552  104001          5$:     ERROR   1               ;RESET ALTERED THE PSW
12445
12446  044554  005037  177776  6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
12447  044560  042737  000004  177564  BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12448
```

```
12449                                   ;;*****************************************************************
12450                                   ;*TEST 645       RESET TEST - N:C = 0000
12451                                   ;;*****************************************************************
12452  044566                           TST645:
12453  044566  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12454  044570  012700  000645                  MOV     #645,R0         ;;LOAD R0 WITH TEST NUMBER
12455  044574  013701  044624                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12456  044600  012737  000001  001110          MOV     #1,@#$TIMES     ;NO ITERATIONS ON THIS TEST
12457  044606  012702  177564                  MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12458  044612  005037  177776                  CLR     @#PSW           ;MAKE PRTY. BITS ALL 0'S
12459  044616  052712  000004                  BIS     #4,(R2)         ;SET THE DL11 MAINT. BIT
12460  044622  000257                          CCC                     ;N:C = 0000
12461
12462  044624  000005                  2$:     RESET                   ;TEST THE RESET - IT SHOULD CLEAR THE DL11 MAINT BIT
12463
12464  044626  013705  177776                  MOV     @#PSW,R5        ;SAVE THE PSW
12465  044632  032712  000004                  BIT     #4,(R2)         ;DID MAINT. BIT CLEAR ??
12466  044636  001403                          BEQ     4$              ;BR IF YES
12467
12468  044640  042712  000004                  BIC     #4,(R2)         ;MAKE SURE TO TURN OFF MAINT. BIT
12469  044644  104006                  3$:     ERROR   6               ;RESET FAILED TO CLEAR MAINT BIT
12470
12471  044646  022705  000000          4$:     CMP     #0,R5           ;DID RESET ALTER THE PSW ??
12472  044652  001406                          BEQ     6$              ;BR IF NOT
12473
12474  044654  012704  000357                  MOV     #357,R4         ;[R4] = S/B PSW
12475  044660  010503                          MOV     R5,R3           ;[R3] = WAS PSW
12476  044662  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12477  044666  104001                  5$:     ERROR   1               ;RESET ALTERED THE PSW
12478
12479  044670  005037  177776          6$:     CLR     @#PSW           ;CLEAR OUT THE PSW
12480  044674  042737  000004  177564          BIC     #4,@#XCSR       ;MAKE SURE MAINT BIT IS OFF
12481
12482                                   ;;*****************************************************************
12483                                   ;*TEST 646       WAIT INSTRUCTION TEST - [PSW] = 151
12484                                   ;;*****************************************************************
12485  044702                           TST646:
12486  044702  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12487  044704  012700  000646                  MOV     #646,R0         ;;LOAD R0 WITH TEST NUMBER
12488  044710  013701  044776                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12489  044714  010605                          MOV     SP,R5           ;SAVE THE SP
12490  044716  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12491  044722  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12492  044726  012737  045014  000064          MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12493  044734  012737  000200  000066          MOV     #200,@#66       ;AT LEVEL 4
12494  044742  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12495  044744  005012                          CLR     (R2)            ;INIT DL11 XCSR
12496  044746  005003                          CLR     R3              ;INIT TIMER
12497
12498  044750  105712                  3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12499  044752  100403                          BMI     5$              ;BR IF YES
12500  044754  005303                          DEC     R3              ;COUNT THE TIMER
12501  044756  001374                          BNE     3$              ;BR IF NO TIMEOUT
12502  044760  000440                          BR      9$              ;GO REPORT TIMEOUT
12503
12504  044762  012737  0C0140  177776  5$:     MOV     #140,@#PSW      ;SET PSW PRTY BITS TO LEVEL 3
```

```
12505  044770  000277                          SCC                     ;N:C=1111
12506  044772  152712  000100                  BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1001
12507
12508  044776  000001                  2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12509
12510  045000  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
12511  045006  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12512  045010  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12513  045012  000424                          BR      8$              ;GO EXIT THIS TEST
12514
12515  045014  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12516  045020  022716  045000                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12517  045024  001402                          BEQ     6$              ;BR IF YES
12518
12519  045026  104006                          ERROR   6               ;WAIT NOT FETCHED PROPERLY
12520  045030  000415                          BR      8$              ;GO EXIT THE TEST
12521
12522  045032  022766  000151  000002  6$:     CMP     #151,2(SP)      ;DID "WAIT" ALTER THE PSW ??
12523  045040  001411                          BEQ     8$              ;BR IF YES
12524
12525  045042  012704  000151                  MOV     #151,R4         ;[R4] = S/B PSW
12526  045046  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
12527  045052  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12528  045056  104001                  7$:     ERROR   1               ;"WAIT" ALTERED THE PSW
12529  045060  000401                          BR      8$              ;GOT TO EXIT TEST
12530
12531  045062  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12532
12533  045064  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12534  045066  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
12535  045072  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
12536  045074  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12537  045102  005037  000066                  CLR     @#66
12538
12539                                  ;;*******************************************************************
12540                                  ;*TEST 647      WAIT INSTRUCTION TEST - [PSW] = 010
12541                                  ;;*************************************************** ************
12542  045106                          TST647:
12543  045106  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12544  045110  012700  000647                  MOV     #647,R0         ;;LOAD R0 WITH TEST NUMBER
12545  045114  013701  045200                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12546  045120  010605                          MOV     SP,R5           ;SAVE THE SP
12547  045122  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12548  045126  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINT TO DL11 XCSR
12549  045132  012737  045216  000064          MOV     #4$,@#64        ;GO TO 4$ ON DL11 INTR.
12550  045140  012737  000200  000066          MOV     #200,@#66       ;AT LEVEL 4
12551  045146  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12552  045150  005012                          CLR     (R2)            ;INIT DL11 XCSR
12553  045152  005003                          CLR     R3              ;INIT TIMER
12554
12555  045154  105712                  3$:     TSTB    (R2)            ;DL11 XMIT READY SET ??
12556  045156  100403                          BMI     5$              ;BR IF YES
12557  045160  005303                          DEC     R3              ;COUNT THE TIMER
12558  045162  001374                          BNE     3$              ;BR IF NO TIMEOUT
12559  045164  000437                          BR      9$              ;GO REPORT TIMEOUT
12560
```

```
12561  045166  005037  177776          5$:     CLR     @#PSW           ;SET PSW PRTY BITS TO LEVEL 0
12562  045172  000257                          CCC                     ;N:C=0000
12563  045174  152712  000100                  BISB    #100,(R2)       ;ENAB. DL11 INTR - N:C=1000
12564
12565  045200  000001                  2$:     WAIT                    ;TEST THE WAIT-GO TO 4$ ON INTR
12566
12567  045202  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT INTR
12568  045210  005012                          CLR     (R2)            ;TURN OFF DL11 INTR ENAB
12569  045212  104006                          ERROR   6               ;WAIT FAILED TO EXECUTE PROPERLY
12570  045214  000424                          BR      8$              ;GO EXIT THIS TEST
12571
12572  045216  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF DL11 INTR ENAB
12573  045222  022716  045202                  CMP     #2$+2,(SP)      ;DID WAIT GET FETCHED ??
12574  045226  001402                          BEQ     6$              ;BR IF YES
12575
12576  045230  104006                          ERROR   6               ;WAIT NOT FETCHED PROPERLY
12577  045232  000415                          BR      8$              ;GO EXIT THE TEST
12578
12579  045234  022766  000010  000002  6$:     CMP     #010,2(SP)      ;DID "WAIT" ALTER THE PSW ??
12580  045242  001411                          BEQ     8$              ;BR IF NO
12581
12582  045244  012704  000010                  MOV     #010,R4         ;[R4] = S/B PSW
12583  045250  016603  000002                  MOV     2(SP),R3        ;[R3] = WAS PSW
12584  045254  012702  177776                  MOV     #PSW,R2         ;DEST = PSW
12585  045260  104001                  7$:     ERROR   1               ;"WAIT" ALTERED THE PSW
12586  045262  000401                          BR      8$              ;GOT TO EXIT TEST
12587
12588  045264  104006                  9$:     ERROR   6               ;DL11 FAILED TO SET READY ON TIME
12589
12590  045266  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12591  045270  005037  177776                  CLR     @#PSW           ;CLEAR OUT THE PSW
12592  045274  005012                          CLR     (R2)            ;TURN OFF DL11 INTR.
12593  045276  012737  000066  000064          MOV     #66,@#64        ;RESTORE DL11 VECTOR WITH TRAPCATCHER
12594  045304  005037  000066                  CLR     @#66
12595
12596                                  ;;*******************************************************************
12597                                  ;*TEST 650      BR PRIORITY ARBITRATION TEST - LEVEL 1 USING LINE CLK
12598                                  ;;*******************************************************************
12599  045310                          TST650:
12600  045310  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12601  045312  012700  000650                  MOV     #650,R0         ;;LOAD R0 WITH TEST NUMBER
12602  045316  013701  045364                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12603  045322  010605                          MOV     SP,R5           ;SAVE THE SP
12604  045324  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12605  045330  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12606  045334  012737  045402  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12607  045342  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12608  045350  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12609  045352  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12610  045354  012737  000040  177776          MOV     #40,@#PSW       ;SET CPU PRIORITY TO LEVEL 1
12611  045362  000257                          CCC                     ;SCOPE SYNC
12612
12613  045364  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12614
12615  045370  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12616  045372  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
```

```
12617
12618  045374  042712  000100          BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12619  045400  104006          3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 1
12620
12621  045402  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
```

```
12622  045406  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12623  045414  005037  000102                  CLR     @#102
12624  045420  010506                          MOV     R5,SP           ;RESET THE SP
12625  045422  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12626
12627                                  ;;*********************************************************************
12628                                  ;*TEST 651       BR PRIORITY ARBITRATION TEST - LEVEL 2 USING LINE CLK
12629                                  ;;*********************************************************************
12630  045426                          TST651:
12631  045426  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12632  045430  012700  000651                  MOV     #651,R0         ;;LOAD R0 WITH TEST NUMBER
12633  045434  013701  045502                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12634  045440  010605                          MOV     SP,R5           ;SAVE THE SP
12635  045442  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12636  045446  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12637  045452  012737  045520  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12638  045460  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12639  045466  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12640  045470  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12641  045472  012737  000100  177776          MOV     #100,@#PSW      ;SET CPU PRIORITY TO LEVEL 2
12642  045500  000257                          CCC                     ;SCOPE SYNC
12643
12644  045502  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12645
12646  045506  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12647  045510  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12648
12649  045512  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12650  045516  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 2
12651
:2652  045520  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12653  045524  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12654  045532  005037  000102                  CLR     @#102
12655  045536  010506                          MOV     R5,SP           ;RESET THE SP
12656  045540  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12657
12658                                  ;;*********************************************************************
12659                                  ;*TEST 652       BR PRIORITY ARBITRATION TEST - LEVEL 3 USING LINE CLK
12660                                  ;;*********************************************************************
12661  045544                          TST652:
12662  045544  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12663  045546  012700  000652                  MOV     #652,R0         ;;LOAD R0 WITH TEST NUMBER
12664  045552  013701  045620                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12665  045556  010605                          MOV     SP,R5           ;SAVE THE SP
12666  045560  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12667  045564  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12668  045570  012737  045636  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12669  045576  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12670  045604  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12671  045606  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12672  045610  012737  000140  177776          MOV     #140,@#PSW      ;SET CPU PRIORITY TO LEVEL 3
12673  045616  000257                          CCC                     ;SCOPE SYNC
12674
12675  045620  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12676
12677  045624  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
```

```
12678  045626  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12679
12680  045630  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12681  045634  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 3
12682
12683  045636  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12684  045642  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12685  045650  005037  000102                  CLR     @#102
12686  045654  010506                          MOV     R5,SP           ;RESET THE SP
12687  045656  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12688
12689                                  ;;*******************************************************************
12690                                  ;*TEST 653      BR PRIORITY ARBITRATION TEST - LEVEL 4 USING LINE CLK
12691                                  ;;*******************************************************************
12692  045662                          TST653:
12693  045662  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12694  045664  012700  000653                  MOV     #653,R0         ;;LOAD R0 WITH TEST NUMBER
12695  045670  013701  045736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12696  045674  010605                          MOV     SP,R5           ;SAVE THE SP
12697  045676  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12698  045702  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12699  045706  012737  045754  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12700  045714  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12701  045722  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12702  045724  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12703  045726  012737  000200  177776          MOV     #200,@#PSW      ;SET CPU PRIORITY TO LEVEL 4
12704  045734  000257                          CCC                     ;SCOPE SYNC
12705
12706  045736  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12707
12708  045742  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12709  045744  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12710
12711  045746  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12712  045752  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 4
12713
12714  045754  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12715  045760  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12716  045766  005037  000102                  CLR     @#102
12717  045772  010506                          MOV     R5,SP           ;RESET THE SP
12718  045774  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12719
12720                                  ;;*******************************************************************
12721                                  ;*TEST 654      BR PRIOPITY ARBITRATION TEST - LEVEL 5 USING LINE CLK
12722                                  ;;*******************************************************************
12723  046000                          TST654:
12724  046000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12725  046002  012700  000654                  MOV     #654,R0         ;;LOAD R0 WITH TEST NUMBER
12726  046006  013701  046054                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12727  046012  010605                          MOV     SP,R5           ;SAVE THE SP
12728  046014  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12729  046020  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12730  046024  012737  046072  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12731  046032  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12732  046040  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12733  046042  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
```

```
12734  046044  012737  000240  177776          MOV     #240,@#PSW      ;SET CPU PRIORITY TO LEVEL 5
12735  046052  000257                          CCC                     ;SCOPE SYNC
12736
12737  046054  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE LINE CLK INTERRUPTS
12738
12739  046060  005304                          DEC     R4              ;COUNT THE TIMER - LCLK SHOULD PREVENT
12740  046062  001376                          BNE     .-2             ;TIMER FROM GETTING BACK TO 000000
12741
12742  046064  042712  000100                  BIC     #100,(R2)       ;TURN OFF THE INTERRUPT ENABLE
12743  046070  104006                  3$:     ERROR   6               ;LINE CLK FAILED TO INTR AT LEVEL 5
12744
12745  046072  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF INTR. ENABLE
12746  046076  012737  000102  000100          MOV     #102,@#100      ;RESTORE TRAP CATCHER IN THE VECTOR
12747  046104  005037  000102                  CLR     @#102
12748  046110  010506                          MOV     R5,SP           ;RESET THE SP
12749  046112  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12750
12751                                  ;;*******************************************************************
12752                                  ;*TEST 655      BR PRIORITY ARBITRATION TEST - LEVEL 6 USING LINE CLK
12753                                  ;;*******************************************************************
12754  046116                          TST655:
12755  046116  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12756  046120  012700  000655                  MOV     #655,R0         ;;LOAD R0 WITH TEST NUMBER
12757  046124  013701  046204                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12758                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT13
12759  046130  032737  020000  063234          BIT     #BIT13,@#BPTLOC ;BREAKPOINT HALT SET ??
12760  046136  001401                          BEQ     .+4             ;BR IF NOT
12761  046140  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
12762  046142  010605                          MOV     SP,R5           ;SAVE THE SP
12763  046144  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12764  046150  012702  177546          1$:     MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12765  046154  012737  046216  000100          MOV     #4$,@#100       ;IF INTR OCCURS - GO TO 4$
12766  046162  012737  000340  000102          MOV     #340,@#102      ;WITH CPU PRIORITY AT LEVEL 7
12767  046170  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12768  046172  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12769  046174  012737  000300  177776          MOV     #300,@#PSW      ;SET CPU PRIORITY TO LEVEL 6
12770  046202  000257                          CCC                     ;SCOPE SYNC
12771
12772  046204  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12773
12774  046210  005304                          DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12775  046212  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12776  046214  000403                          BR      6$              ;GO TO EXIT - ALL OK
12777
12778  046216  042712  000100          4$:     BIC     #1C0,(R2)       ;TURN OFF THE INTR ENABLE
12779  046222  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 6
12780
12781  046224  042712  000100          6$:     BIC     #1C0,(R2)       ;TURN OFF INTR ENABLE
12782  046230  012737  000102  000100          MOV     #1C2,@#100      ;RESET THE TRAP CATCHER IN THE VECTOR
12783  046236  005037  000102                  CLR     @#102
12784  046242  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12785  046244  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12786
12787                                  ;;*******************************************************************
12788                                  ;*TEST 656      BR PRIORITY ARBITRATION TEST - LEVEL 7 USING DL11
12789                                  ;;*******************************************************************
```

```
12790  046250                          TST656:
12791  046250  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12792  046252  012700  000656                  MOV     #656,R0         ;;LOAD R0 WITH TEST NUMBER
12793  046256  013701  046324                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12794  046262  010605                          MOV     SP,R5           ;SAVE THE SP
12795  046264  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12796  046270  012702  177564          1$:     MOV     #XCSR,R2        ;R2 POINTS TO DL11 XCSR
12797  046274  012737  046336  000064          MOV     #4$,@#64        ;IF INTR OCCURS - GO TO 4$
12798  046302  012737  000340  000066          MOV     #340,@#66       ;WITH CPU PRIORITY AT LEVEL 7
12799  046310  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
12800  046312  005004                          CLR     R4              ;INITIALIZE R4 AS TIMER
12801  046314  012737  000340  177776          MOV     #340,@#PSW      ;SET CPU PRIORITY TO LEVEL 7
12802  046322  000257                          CCC                     ;SCOPE SYNC
12803
12804  046324  052712  000100          2$:     BIS     #100,(R2)       ;ENABLE INTERRUPTS
12805
12806  046330  005304                          DEC     R4              ;COUNT UNTIL [R4] = 000000 - THEN
12807  046332  001376                          BNE     .-2             ;CONTINUE - NO INTERRUPT SHOULD OCCUR
12808  046334  000403                          BR      6$              ;GO TO EXIT - ALL OK
12809
12810  046336  042712  000100          4$:     BIC     #100,(R2)       ;TURN OFF THE INTR ENABLE
12811  046342  104006                  3$:     ERROR   6               ;INTR OCCURRED WITH CPU AT LEVEL 7
12812
12813  046344  042712  000100          6$:     BIC     #100,(R2)       ;TURN OFF INTR ENABLE
12814  046350  012737  000066  000064          MOV     #66,@#64        ;RESET THE TRAP CATCHER IN THE VECTOR
12815  046356  005037  000066                  CLR     @#66
12816  046362  010506                          MOV     R5,SP           ;RESET SP JUST IN CASE
12817  046364  005037  177776                  CLR     @#PSW           ;SET CPU PRIORITY BACK TO LEVEL 0
12818
12819                                  ;;**************************************************************
12820                                  ;*TEST 657      "CLR  @#PSW" ALLOWS IMMEDIATE BR-BG-INTR SEQUENCE
12821                                  ;THIS TEST VERIFIES THAT IF A "BR" REQUEST IS PENDING WHEN A "CLR  @#PSW"
12822                                  ;IS EXECUTED TO LOWER THE CPU PRIORITY, THE REQUEST IS GRANTED BEFORE
12823                                  ;EXECUTION OF THE INSTRUCTION FOLLOWING THE "CLR"
12824                                  ;;**************************************************************
12825  046370                          TST657:
12826  046370  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12827  046372  012700  000657                  MOV     #657,R0         ;;LOAD R0 WITH TEST NUMBER
12828  046376  013701  046470                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12829  046402  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12830  046406  010605                          MOV     SP,R5           ;SAVE THE SP
12831  046410  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12832  046414  012737  046476  000100  1$:     MOV     #4$,@#100       ;SET UP LCLK VECTOR TO GO TO 4$
12833  046422  012737  000300  000102          MOV     #300,@#102
12834  046430  010506                          MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12835  046432  005004                          CLR     R4              ;INITIALIZE TIMER FO KW
12836  046434  005003                          CLR     R3              ;CLEAR SOFTWARE FLAG
12837  046436  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12838  046444  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12839  046450  042712  000200                  BIC     #200,(R2)       ;CLEAR LINE CLOCK READY
12840  046454  105712                  11$:    TSTB    (R2)            ;LCLK READY TO INTR ??
12841  046456  100403                          BMI     12$             ;BR IF YES
12842  046460  005304                          DEC     R4              ;COUNT THE TIMER
12843  046462  001374                          BNE     11$             ;BR IF NO TIMEOUT
12844  046464  000411                          BR      6$              ;GO REPORT TIMEOUT
12845  046466  000257                  12$:    CCC                     ;SCOPE SYNC
```

```
12846
12847  046470  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - LCLK SHOULD INTERRUPT
12848                                                                   ;BEFORE FETCHING NEXT INSTRUCTION
12849  046474  005103                          COM     R3              ;SHOULD NOT BE FETCHED
12850  046476  005012                  4$:     CLR     (R2)            ;DISABLE THE LCLK INTR
12851  046500  005703                          TST     R3              ;DID SOFTWARE FLAG GET SET ??
12852  046502  001404                          BEQ     8$              ;BR IF NOT - IT WORKED OK
12853  046504  104006                  3$:     ERROR   6               ;LCLK FAILED TO INTR ONTIME
12854  046506  000402                          BR      8$              ;GO EXIT
12855
12856  046510  005012                  6$:     CLR     (R2)            ;DISABLE  LCLK INTR
12857  046512  104006                  5$:     ERROR   6               ;LINE CLK TIMED OUT
12858
12859  046514  010506                  8$:     MOV     R5,SP           ;RESET THE SP
12860  046516  012737  000102  000100          MOV     #102,@#100      ;RESTORE THE LINE CLK TRAPCATCHER
12861  046524  005037  000102                  CLR     @#102
12862
12863                                  ;;*****************************************************************
12864                                  ;*TEST 660      "BR6 VS BR4" PRIORITY ARBITRATION TEST
12865                                  ;THIS TEST VERIFIES THAT IF BOTH A "BR4" AND A "BR6" REQUEST ARE
12866                                  ;PENDING WHEN THE CPU PRIORITY IS LOWERED TO ALLOW INTRS. THAT "BR6"
12867                                  ;REQUEST IS GRANTED FIRST EVEN THOUGH THE "BR4" REQUEST MAY HAVE
12868                                  ;OCCURRED FIRST
12869                                  ;;*****************************************************************
12870  046530                          TST660:
12871  046530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12872  046532  012700  000660                  MOV     #660,R0         ;;LOAD R0 WITH TEST NUMBER
12873  046536  013701  046676                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12874  046542  010605                          MOV     SP,R5           ;SAVE THE SP
12875  046544  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12876  046550  012702  177546                  MOV     #LKCSR,R2       ;R2 POINTS TO LINE CLK CSR
12877  046554  012703  177564                  MOV     #XCSR,R3        ;R3 POINTS TO DL11 XCSR
12878  046560  012737  046706  000100          MOV     #4$,@#100       ;SET UP THE LCLK VECTOR - GO TO 4$
12879  046566  012737  000300  000102          MOV     #300,@#102
12880  046574  012737  046740  000064          MOV     #8$,@#64        ;SET UP THE DL11 VECTOR - GO TO 8$
12881  046602  012737  000200  000066          MOV     #200,@#66
12882  046610  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOPING
12883  046612  012737  000340  177776          MOV     #340,@#PSW      ;LOCK OUT ALL INTRS
12884  046620  005037  063312                  CLR     @#MBUF0         ;INIT TIMER
12885  046624  005037  063316                  CLR     @#MBUF1         ;CLEAR DL11 INTR FLAG
12886  046630  005004                          CLR     R4              ;INIT TIMER
12887  046632  052713  000100                  BIS     #100,(R3)       ;ENABLE DL11 XMIT INTR
12888  046636  105713                  11$:    TSTB    (R3)            ;XMIT READY SET ??
12889  046640  100403                          BMI     12$             ;BR IF YES
12890  046642  005304                          DEC     R4              ;COUNT THE TIMER
12891  046644  001374                          BNE     11$             ;BR IF NO TIMEOUT
12892  046646  000443                          BR      5$              ;GO REPORT TIMEOUT FOR DL11
12893
12894  046650  005004                  12$:    CLR     R4              ;INIT THE TIMER AGAIN
12895  046652  052712  000100                  BIS     #100,(R2)       ;ENABLE LCLK INTRS
12896  046656  042712  000200                  BIC     #200,(R2)       ;CLEAR THE LINE CLOCK READY BIT
12897  046662  105712                  13$:    TSTB    (R2)            ;LCLK READY TO INTR
12898  046664  100403                          BMI     14$             ;BR IF YES
12899  046666  005304                          DEC     R4              ;COUNT THE TIMER
12900  046670  001374                          BNE     13$             ;BR IF NO TIMEOUT
12901  046672  000436                          BR      7$              ;GO REPORT LINE CLK TIMEOUT
```

```
12902  046674  000257                  14$:    CCC                     ;SCOPE SYNC
12903
12904  046676  005037  177776          2$:     CLR     @#PSW           ;ALLOW INTRS - KW SHOULD INTR FIRST
12905
12906  046702  005137  063312                  COM     @#MBUF0         ;SET SOFTWARE FLAG IF FETCHED
12907  046706  005013                  4$:     CLR     (R3)            ;DISABLE BOTH INTERRUPTS
12908  046710  005012                          CLR     (R2)
12909  046712  005737  063312                  TST     @#MBUF0         ;DID SOFTWARE FLAG GET SET ??
12910  046716  001402                          BEQ     6$              ;BR IF NOT
12911
12912  046720  104006                  3$:     ERROR   6               ;LINE CLK INTR OCCURRED TOO LATE
12913  046722  000425                          BR      9$              ;GO TO EXIT
12914
12915  046724  005737  063316          6$:     TST     @#MBUF1         ;DID DL11 SOFTWARE FLAG SET ??
12916  046730  001422                          BEQ     9$              ;BR IF NOT
12917
12918  046732  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12919  046734  104006                          ERROR   6               ;DL11 INTERRUPTED THE KW11
12920  046736  000417                          BR      9$              ;GO TO EXIT TEST
12921
12922  046740  005137  063316          8$:     COM     @#MBUF1         ;FLAG THE DL11 INTR
12923  046744  005013                          CLR     (R3)            ;DISABLE BOTH INTR ENABLES
12924  046746  005012                          CLR     (R2)
12925  046750  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12926  046752  104006                          ERROR   6               ;DL11 SHOULD NOT HAVE INTERRUPTED
12927  046754  000410                          BR      9$              ;GO EXIT TEST
12928
12929  046756  005012                  5$:     CLR     (R2)            ;DISABLE THE INTR ENABLES
12930  046760  005013                          CLR     (R3)
12931  046762  010302                          MOV     R3,R2           ;FOR CORRECT DESTINATION TYPOUT
12932  046764  104006                          ERROR   6               ;DL11 TIMEOUT
12933  046766  000403                          BR      9$              ;GO TO EXIT
12934
12935  046770  005012                  7$:     CLR     (R2)            ;DISABLE INTR ENABLES
12936  046772  005013                          CLR     (R3)
12937  046774  104006                          ERROR   6               ;KW11 TIMEOUT
12938
12939  046776  010506                  9$:     MOV     R5,SP           ;RESET THE SP
12940  047000  005037  177776                  CLR     @#PSW           ;RESET THE CPU PRIORITY
12941  047004  012737  000102  000100          MOV     #102,@#100      ;RESTORE LCLK VECTOR
12942  047012  005037  000102                  CLR     @#102
12943  047016  012737  000066  000064          MOV     #66,@#64        ;RESTORE THE DL11 XMIT VECTOR
12944  047024  005037  000066                  CLR     @#66
12945
12946                                  ; ************************************************************************
12947                                  ; //////////////COMBINED INSTRUCTION EXERCISER SECTION //////////////
12948                                  ; ************************************************************************
12949
12950                                  ;;*********************************************************
12951                                  ;*TEST 661       "BPT" TRAP LINKAGE TEST
12952                                  ;;*********************************************************
12953  047030                  TST661:
12954  047030  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12955  047032  012700  000661                  MOV     #661,R0         ;;LOAD R0 WITH TEST NUMBER
12956  047036  013701  047062                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12957  047042  010605                          MOV     SP,R5           ;SAVE THE SP
```

```
12958  047044  010737  001010                 MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
12959  047050  012737  047066  000014  1$:    MOV     #4$,@#14        ;GO TO 4$ ON "BPT" TRAP
12960  047056  010506                         MOV     R5,SP           ;RESET THE SP FOR ERROR LOOPING
12961  047060  000257                         CCC                     ;SCOPE SYNC
12962
12963  047062  000003                 2$:     BPT                     ;TEST THE "BPT" - GO TO 4$
12964
12965  047064  104005                 3$:     ERROR   5               ;BPT FAILED TO TRAP
12966
12967  047066  010506                 4$:     MOV     R5,SP           ;RESET THE SP
12968  047070  012737  000016  000014         MOV     #16,@#14        ;RESTORE THE VECTOR
12969
12970                                 ;;*****************************************************************
12971                                 ;*TEST 662       RED ZONE OVERFLOW TEST - MOV R,-(SP)
12972                                 ;;*****************************************************************
12973  047076                         TST662:
12974  047076  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
12975  047100  012700  000662                 MOV     #662,R0         ;;LOAD R0 WITH TEST NUMBER
12976  047104  013701  047144                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
12977  047110  010605                         MOV     SP,R5           ;SAVE SP
12978  047112  013704  000004                 MOV     @#4,R4          ;SAVE T.O. VECTOR
12979  047116  013703  000336                 MOV     @#336,R3        ;SAVE VECTOR AT 336
12980  047122  012737  047162  000004         MOV     #4$,@#4 ;GO TO 4$ ON OVFLW
12981  047130  012737  125252  000336         MOV     #125252,@#336   ;INIT. [336]
12982  047136  012706  000340                 MOV     #340,SP         ;SET SP TO CAUSE RED ZONE TRAP
12983  047142  000257                         CCC                     ;SCOPE SYNC
12984
12985  047144  010046                 2$:     MOV     R0,-(SP)        ;FORCE RED ZONE TRAP - GO TO 4$
12986
12987  047146  010437  000004                 MOV     R4,@#4          ;RESTORE T.O. VECTOR
12988  047152  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12989  047156  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
12990  047160  104005                 3$:     ERROR   5               ;MOV FAILED TO CAUSE TRAP
12991
12992  047162  010437  000004         4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
12993  047166  022706  000000                 CMP     #0,SP           ;[SP]=0?
12994  047172  001404                         BEQ     6$              ;BE IF YES
12995
12996  047174  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
12997  047200  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
12998  047202  104005                 5$:     ERROR   5               ;SP NOT BEING JAMMED TO 4
12999
13000  047204  022737  125252  000336  6$:     CMP     #125252,@#336   ;DID PUSH OCCUR IN YELLOW ZONE?
13001  047212  001404                         BEQ     8$              ;BR IF NOT
13002
13003  047214  010637  001074                 MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13004  047220  010506                         MOV     R5,SP           ;RESET SP FOR ERROR CALL
13005  047222  104005                 7$:     ERROR   5               ;MOV PUSHED INTO YELLOW ZONE
13006
13007  047224  010337  000336         8$:     MOV     R3,@#336        ;RESTORE VECTOR 336
13008  047230  010506                         MOV     R5,SP           ;RESET SP
13009
13010                                 ;;*****************************************************************
13011                                 ;*TEST 663       YELLOW ZONE OVERFLOW TEST - MOV R,-(SP)
13012                                 ;;*****************************************************************
13013  047232                         TST663:
```

```
13014  047232  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13015  047234  012700  000663                  MOV     #663,R0         ;;LOAD R0 WITH TEST NUMBER
13016  047240  013701  047272                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13017  047244  010605                          MOV     SP,R5           ;SAVE SP
13018  047246  012702  000376                  MOV     #376,R2         ;R2 POINTS TO STACK
13019  047252  013704  000004                  MOV     @#4,R4          ;SAVE T.O. VECTOR
13020  047256  012737  047310  000004          MOV     #4$,@#4 ;ON OVFLW - GO TO 4$
13021  047264  012706  000400                  MOV     #400,SP         ;SET SP TO CAUSE OVFLW
13022  047270  000257                          CCC                     ;SCOPE SYNC
13023
13024  047272  010046                  2$:     MOV     R0,-(SP)        ;FORCE STACK OVFLW - GO TO 4$
13025
13026  047274  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13027  047300  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13028  047304  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13029  047306  104005                  3$:     ERROR   5               ;STACK OVFLW FAILED TO TRAP
13030
13031  047310  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13032  047314  020012                          CMP     R0,(R2)         ;DID [R0] GET PUSHED?
13033  047316  001404                          BEQ     6$              ;BR IF YES
13034
13035  047320  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13036  047324  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13037  047326  104005                  5$:     ERROR   5               ;MOV FAILED TO PUSH IN YELLOW ZONE
13038
13039  047330  005706                  6$:     TST     SP              ;[SP]=0?
13040  047332  001004                          BNE     8$              ;BR IF NOT
13041
13042  047334  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13043  047340  010506                          MOV     R5,SP           ;RESET SP FOR ERROR CALL
13044  047342  104005                  7$:     ERROR   5               ;RED ZONE INSTEAD OF YELLOW ZONE
13045
13046  047344  010506                  8$:     MOV     R5,SP           ;RESET SP
13047
13048                                  ;;**************************************************
13049                                  ;*TEST 664       YELLOW ZONE OVERFLOW TEST - (CMP  R0,-(SP))
13050                                  ;;************************************ *************
13051  047346                          TST664:
13052  047346  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13053  047350  012700  000664                  MOV     #664,R0         ;;LOAD R0 WITH TEST NUMBER
13054  047354  013701  047402                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13055  047360  010605                          MOV     SP,R5           ;SAVE THE SP
13056  047362  013704  000004                  MOV     @#4,R4          ;SAVE TRAP VECTOR
13057  047366  012737  047406  000004          MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13058  047374  012706  000400                  MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13059  047400  000257                          CCC                     ;SCOPE SYNC
13060
13061  047402  020046                  2$:     CMP     R0,-(SP)        ;TEST THE CMP - NO TRAP SHOULD OCCUR
13062
13063  047404  000406                          BR      6$              ;GO TO EXIT TEST
13064
13065  047406  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13066  047412  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13067  047416  010506                          MOV     R5,SP           ;RESET THE SP
13068  047420  104005                  3$:     ERROR   5               ;CMP CAUSED OVERFLOW TRAP
13069
```

```
13070  047422  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13071  047426  010506                          MOV     R5,SP           ;RESET THE SP
13072
13073                                  ;;*****************************************************************
13074                                  ;*TEST 665      YELLOW ZONE OVERFLOW TEST - (BIT  R0,-(SP))
13075                                  ;;*****************************************************************
13076  047430                          TST665:
13077  047430  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13078  047432  012700  000665                 MOV     #665,R0         ;;LOAD R0 WITH TEST NUMBER
13079  047436  013701  047464                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13080  047442  010605                          MOV     SP,R5           ;SAVE THE SP
13081  047444  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13082  047450  012737  047470  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13083  047456  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13084  047462  000257                         CCC                     ;SCOPE SYNC
13085
13086  047464  030046                  2$:     BIT     R0,-(SP)        ;TEST THE BIT - NO TRAP SHOULD OCCUR
13087
13088  047466  000406                          BR      6$              ;GO TO EXIT TEST
13089
13090  047470  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13091  047474  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13092  047500  010506                          MOV     R5,SP           ;RESET THE SP
13093  047502  104005                  3$:     ERROR   5               ;BIT CAUSED OVERFLOW TRAP
13094
13095  047504  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13096  047510  010506                          MOV     R5,SP           ;RESET THE SP
13097
13098                                  ;;*****************************************************************
13099                                  ;*TEST 666      YELLOW ZONE OVERFLOW TEST - (TST  -(SP))
13100                                  ;;*****************************************************************
13101  047512                          TST666:
13102  047512  000004                         SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13103  047514  012700  000666                 MOV     #666,R0         ;;LOAD R0 WITH TEST NUMBER
13104  047520  013701  047546                 MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13105  047524  010605                          MOV     SP,R5           ;SAVE THE SP
13106  047526  013704  000004                 MOV     @#4,R4          ;SAVE TRAP VECTOR
13107  047532  012737  047552  000004         MOV     #4$,@#4 ;GO TO 4$ IF TRAP SPRUNG
13108  047540  012706  000400                 MOV     #400,SP         ;SET SP TO PUSH INTO "YELLOW ZONE"
13109  047544  000257                         CCC                     ;SCOPE SYNC
13110
13111  047546  005746                  2$:     TST     -(SP)           ;TEST THE TST - NO TRAP SHOULD OCCUR
13112
13113  047550  000406                          BR      6$              ;GO TO EXIT TEST
13114
13115  047552  010437  000004          4$:     MOV     R4,@#4          ;RESTORE TRAP VECTOR
13116  047556  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13117  047562  010506                          MOV     R5,SP           ;RESET THE SP
13118  047564  104006                  3$:     ERROR   6               ;TST CAUSED OVERFLOW TRAP
13119
13120  047566  010437  000004          6$:     MOV     R4,@#4          ;RESTORE THE VECTOR
13121  047572  010506                          MOV     R5,SP           ;RESET THE SP
13122
13123                                  ;;*****************************************************************
13124                                  ;*TEST 667      ODD ADDRESS ERROR TEST - SUB  RA,(RB) - (RB) = ODD
13125                                  ;;*****************************************************************
```

```
13126  047574                          TST667:
13127  047574  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13128  047576  012700  000667                  MOV     #667,R0         ;;LOAD R0 WITH TEST NUMBER
13129  047602  013701  047636                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13130  047606  010605                          MOV     SP,R5           ;SAVE SP
13131  047610  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13132  047614  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13133  047620  012737  047646  000004        MOV     #4$,@#4 ;ON ODD ADDR ERROR - GO TO 4$
13134  047626  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13135  047630  012702  000001                  MOV     #1,R2           ;R2 GETS ODD ADDRESS
13136  047634  000257                          CCC                     ;SCOPE SYNC
13137
13138  047636  160012          2$:     SUB     R0,(R2)         ;FORCE ODD ADDR ERROR - GO TO 4$
13139
13140  047640  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13141  047644  104006          3$:     ERROR   6               ;ODD ADDR FAILED TO TRAP
13142
13143  047646  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13144  047652  010506                          MOV     R5,SP           ;RESET SP
13145  047654  005037  000000                  CLR     @#0             ;CLR LOC. 0 JUST IN CASE
13146
13147                                  ;;**********************************************************************
13148                                  ;*TEST 670       TEST FOR ODD ADDR. ERROR TRAP FOR DEST. DEFERRED MODES
13149                                  ;;**********************************************************************
13150  047660                          TST670:
13151  047660  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13152  047662  012700  000670                  MOV     #670,R0         ;;LOAD R0 WITH TEST NUMBER
13153  047666  013701  047710                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13154  047672  012702  063317                  MOV     #MBUF1+1,R2     ;DEST ADDR=MBUF1+1 (ODD)
13155  047676  012737  047760  000004        MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13156
13157  047704  010205                          MOV     R2,R5           ;[R5] = DEST. ADDR
13158  047706  000257                          CCC                     ;SCOPE SYNC
13159
13160  047710  105435          2$:     NEGB    @(R5)+          ;TEST DM=3 TRAP
13161
13162  047712  104006          3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13163
13164  047714  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = DEST. ADDR
13165  047720  013701  047726                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13166  047724  000257                          CCC                     ;SCOPE SYNC
13167
13168  047726  105455          20$:    NEGB    @-(R5)          ;TEST DM=5 TRAP
13169
13170  047730  104006          5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13171
13172  047732  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13173  047734  013701  047742                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13174  047740  000257                          CCC                     ;SCOPE SYNC
13175
13176  047742  105475  000000          21$:    NEGB    @0(R5)          ;TEST DM=7 TRAP
13177
13178  047746  104006          7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13179
13180  047750  012737  061220  000004        MOV     #BERR,@#4       ;RESET T.O. VECTOR
13181  047756  000403                          BR      TST671          ;;GO TO SCOPE EXIT
```

```
13182
13183  047760  062716  000002          4$:     ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13184  047764  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13185
13186                                  ;;*********************************************************************
13187                                  ;*TEST 671       TEST FOR ODD ADDR ERROR TRAP FOR SOURCE DEFERRED MODES
13188                                  ;;*********************************************************************
13189  047766                          TST671:
13190  047766  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13191  047770  012700  000671                  MOV     #671,R0         ;;LOAD R0 WITH TEST NUMBER
13192  047774  013701  050016                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13193  050000  012702  063317                  MOV     #MBUF1+1,R2     ;[R2] = SOURCE ADDR. (ODD)
13194  050004  012737  050066  000004          MOV     #4$,@#4 ;GO TO 4$ ON TRAP
13195
13196  050012  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR.
13197  050014  000257                          CCC                     ;SCOPE SYNC
13198
13199  050016  113504                  2$:     MOVB    @(R5)+,R4       ;TEST SM=3
13200
13201  050020  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13202
13203  050022  012705  063321                  MOV     #MBUF1+3,R5     ;[R5] = SOURCE ADDR
13204  050026  013701  050034                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13205  050032  000257                          CCC                     ;SCOPE SYNC
13206
13207  050034  115504                  20$:    MOVB    @-(R5),R4       ;TEST SM=5
13208
13209  050036  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13210  050040  010205                          MOV     R2,R5           ;[R5] = SOURCE ADDR
13211  050042  013701  050050                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13212  050046  000257                          CCC                     ;SCOPE SYNC
13213
13214  050050  117504  000000          21$:    MOVB    @0(R5),R4       ;TEST SM=7
13215
13216  050054  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13217
13218  050056  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13219  050064  000403                          BR      TST672          ;;GO TO SCOPE EXIT
13220
13221  050066  062716  000002          4$:     ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
13222  050072  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13223
13224                                  ;;*********************************************************************
13225                                  ;*TEST 672       TEST FOR ODD ADDR ERROR TRAP FOR JMP DEST DEFERRED MODES
13226                                  ;;*********************************************************************
13227  050074                          TST672:
13228  050074  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13229  050076  012700  000672                  MOV     #672,R0         ;;LOAD R0 WITH TEST NUMBER
13230  050102  013701  050124                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13231  050106  012702  050177                  MOV     #6$+3,R2        ;DEST ADDR = 6$+3 (ODD)
13232  050112  012737  050202  000004          MOV     #4$,@#4 ;GO TO 4$ ON ODA TRAP
13233
13234  050120  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13235  050122  000257                          CCC                     ;SCOPE SYNC
13236
13237  050124  000135                  2$:     JMP     @(R5)+          ;TEST JMP DM=3
```

```
13238
13239  050126  104006                  3$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 153
13240
13241  050130  012705  050177                  MOV     #6$+3,R5        ;[R5] = DEST ADDR
13242  050134  013701  050142                  MOV     @#20$,R1        ;[R1] = TEST INSTR
13243  050140  000257                          CCC                     ;SCOPE SYNC
13244
13245  050142  000155                  20$:    JMP     @-(R5)          ;TEST JMP DM=5
13246
13247  050144  104006                  5$:     ERROR   6               ;ODA TRAP NOT SPRUNG IN ROM LOC 155
13248
13249  050146  010205                          MOV     R2,R5           ;[R5] = DEST ADDR
13250  050150  013701  050156                  MOV     @#21$,R1        ;[R1] = TEST INSTR
13251  050154  000257                          CCC                     ;SCOPE SYNC
13252
13253  050156  000175  000000          21$:    JMP     @0(R5)          ;TEST JMP DM=7
13254
13255  050162  104006                  7$:     ERROR   6               ;ODA TRAP NOT SPRUNG
13256
13257  050164  012737  061220  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13258  050172  000420                          BR      TST673          ;;GO TO SCOPE EXIT
13259
13260  050174  000000                  6$:     HALT                    ;CATASTOPHIC ERROR - [PC] QUESTIONABLE.
13261  050176  000000                          HALT                    ;RESTART PROGRAM - DO NOT CONTINUE.
13262  050200  000000                          HALT
13263
13264  050202  032716  000001          4$:     BIT     #1,(SP)         ;TRAP DUE TO ODD PC?
13265  050206  001003                          BNE     8$              ;BR IF YES
13266  050210  062716  000002                  ADD     #2,(SP)         ;MOV RETURN PC AROUND ERROR CALL
13267  050214  000002                          RTI                     ;RETURN TO NEXT SUB TEST
13268
13269  050216  011603                  8$:     MOV     (SP),R3         ;GET ODD PC OFF STACK INTO R3
13270  050220  062706  000004                  ADD     #4,SP           ;FIX SP
13271
13272  050224  104007                  9$:     ERROR   7               ;PC TRAPPED WITH ODD ADDRESS
13273
13274  050226  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13275
13276                                  ;;*********************************************************************
13277                                  ;*TEST 673       TEST FOR STACK OFLW FOR DEST MODES 1,2,4, AND 6.
13278                                  ;;*********************************************************************
13279  050234                          TST673:
13280  050234  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13281  050236  012700  000673                  MOV     #673,R0         ;;LOAD R0 WITH TEST NUMBER
13282  050242  013701  050266                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13283  050246  012737  050400  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13284  050254  010605                          MOV     SP,R5           ;SAVE SP
13285  050256  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13286
13287  050262  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13288  050264  000257                          CCC                     ;SCOPE SYNC
13289
13290  050266  005016                  2$:     CLR     (SP)            ;TEST DM1 - SHOULD SPRING TRAP
13291
13292  050270  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13293  050274  010506                          MOV     R5,SP           ;RESET SP
```

```
13294  050276  104006                  3$:     ERROR   6               ;DM1 FAILED TO CAUSE OVERFLOW TRAP
13295
13296  050300  013701  050310                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
13297  050304  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13298  050306  000257                          CCC                     ;SCOPE SYNC
13299
13300  050310  005026                  20$:    CLR     (SP)+           ;TEST DM2 - SHOULD SPRING TRAP
13301
13302  050312  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13303  050316  010506                          MOV     R5,SP           ;RESET SP
13304  050320  104006                  5$:     ERROR   6               ;DM2 FAILED TO CAUSE OVERFLOW TRAP
13305
13306  050322  013701  050332                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
13307  050326  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13308  050330  000257                          CCC                     ;SCOPE SYNC
13309
13310  050332  005046                  21$:    CLR     -(SP)           ;TEST DM4 - SHOULD SPRING TRAP
13311
13312  050334  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13313  050340  010506                          MOV     R5,SP           ;RESET SP
13314  050342  104006                  7$:     ERROR   6               ;DM4 FAILED TO CAUSE OVERFLOW TRAP
13315
13316  050344  013701  050354                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
13317  050350  010206                          MOV     R2,SP           ;SET SP TO CAUSE ERROR
13318  050352  000257                          CCC                     ;SCOPE SYNC
13319
13320  050354  005066  000000          22$:    CLR     0(SP)           ;TEST DM6 - SHOULD SPRING TRAP
13321
13322  050360  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13323  050364  010506                          MOV     R5,SP           ;RESET SP
13324  050366  104006                  9$:     ERROR   6               ;DM6 FAILED TO CAUSE OVERFLOW TRAP
13325
13326  050370  012737  061220  000004          MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13327  050376  000407                          BR      TST674          ;;GO TO SCOPE EXIT
13328
13329  050400  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC OFF STACK
13330  050402  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13331  050406  010506                          MOV     R5,SP           ;RESET SP
13332  050410  005046                          CLR     -(SP)           ,PUSH NEW PS ON STACK
13333  050412  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC ON STACK
13334  050414  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13335
13336                                  ;;**********************************************************************
13337                                  ;*TEST 674       TEST FOR STACK OVFLW FOR MOV DEST MODES 1,2,4, AND 6.
13338                                  ;;**********************************************************************
13339  050416                          TST674:
13340  050416  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13341  050420  012700  000674                  MOV     #674,R0         ;;LOAD R0 WITH TEST NUMBER
13342  050424  013701  050450                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13343  050430  012737  050562  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLW TRAP
13344  050436  010605                          MOV     SP,R5           ;SAVE SP
13345  050440  012702  000376                  MOV     #376,R2         ;USE R2 TO SET UP SP TO CAUSE TRAP
13346
13347  050444  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13348  050446  000257                          CCC                     ;SCOPE SYNC
13349
```

```
13350  050450  010016                  2$:     MOV     R0,(SP)         ;TEST MOV DM1 - SHOULD SPRING TRAP
13351
13352  050452  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13353  050456  010506                          MOV     R5,SP           ;RESET SP
13354  050460  104006                  3$:     ERROR   6               ;MOV DM1 FAILED TO SPRING TRAP
13355
13356  050462  013701  050472                  MOV     @#20$,R1        ;[R1] = TEST INSTR.
13357  050466  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13358  050470  000257                          CCC                     ;SCOPE SYNC
13359
13360  050472  010026                  20$:    MOV     R0,(SP)+        ;TEST MOV DM2 - SHOULD SPRING TRAP
13361
13362  050474  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13363  050500  010506                          MOV     R5,SP           ;RESET SP
13364  050502  104006                  5$:     ERROR   6               ;MOV DM2 FAILED TO SPRING TRAP
13365
13366  050504  013701  050514                  MOV     @#21$,R1        ;[R1] = TEST INSTR.
13367  050510  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13368  050512  000257                          CCC                     ;SCOPE SYNC
13369
13370  050514  010046                  21$:    MOV     R0,-(SP)        ;TEST MOV DM4 - SHOULD SPRING TRAP
13371
13372  050516  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13373  050522  010506                          MOV     R5,SP           ;RESET SP
13374  050524  104006                  7$:     ERROR   6               ;MOV DM4 FAILED TO SPRING TRAP
13375
13376  050526  013701  050536                  MOV     @#22$,R1        ;[R1] = TEST INSTR.
13377  050532  010206                          MOV     R2,SP           ;SET UP SP TO CAUSE OVERFLOW
13378  050534  000257                          CCC                     ;SCOPE SYNC
13379
13380  050536  010066  000000          22$:    MOV     R0,0(SP)        ;TEST MOV DM6 - SHOULD SPRING TRAP
13381
13382  050542  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13383  050546  010506                          MOV     R5,SP           ;RESET SP
13384  050550  104006                  9$:     ERROR   6               ;MOV DM6 FAILED TO CAUSE OVFLW TRAP
13385
13386  050552  012737  061220  000004          MOV     #BERR,@#4       ;RESET T.O. VECTOR
13387  050560  000407                          BR      TST675          ;;GO TO SCOPE EXIT
13388
13389  050562  011604                  4$:     MOV     (SP),R4         ;GET RETURN PC
13390  050564  062704  000010                  ADD     #10,R4          ;MOVE RETURN PC AROUND ERROR CALL
13391  050570  010506                          MOV     R5,SP           ;RESET SP
13392  050572  005046                          CLR     -(SP)           ;PUSH NEW PSW
13393  050574  010446                          MOV     R4,-(SP)        ;PUSH RETURN PC
13394  050576  000002                          RTI                     ;RETURN TO NEXT SUB-TEST
13395
13396                                  ;;*********************************************************
13397                                  ;*TEST 675       TEST THAT JSR CAN CAUSE OVERFLOW TRAP
13398                                  ;;*********************************************************
13399  050600                          TST675:
13400  050600  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13401  050602  012700  C00675                  MOV     #675,R0         ;;LOAD R0 WITH TEST NUMBER
13402  050606  013701  050630                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13403  050612  012737  050652  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVERFLOW ERROR
13404  050620  010605                          MOV     SP,R5           ;SAVE SP
13405  050622  012706  000400                  MOV     #400,SP         ;SET THE SP TO CAUSE TRAP
```

```
13406  050626  000257                          CCC                     ;SCOPE SYNC
13407
13408  050630  004737  050656          2$:     JSR     PC,@#6$ ;TEST JSR - SHOULD SPRING TRAP
13409
13410  050634  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13411  050640  010506                          MOV     R5,SP           ;RESET SP
13412  050642  104005          3$:     ERROR   5                       ;JSR PUSH DID NOT SPRING OVFL TRAP
13413
13414  050644  000410                          BR      8$              ;GO TO SCOPE EXIT
13415
13416  050646  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13417  050652  010506          4$:     MOV     R5,SP           ;RESET SP
13418  050654  000404                          BR      8$              ;GO EXIT TEST - ALL OK
13419
13420  050656  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13421  050662  010506                          MOV     R5,SP           ;RESET SP
13422  050664  104005          5$:     ERROR   5                       ;JSR PUSH FAILED TO SPRING OVFLW TRAP
13423
13424  050666  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET BUS T.O. VECTOR
13425
13426                                  ;;**********************************************************************
13427                                  ;*TEST 676       TEST THAT 1ST PUSH IN TRAP MICROROUTINE CAUSES OVFLW TRAP
13428                                  ;;**********************************************************************
13429  050674                          TST676:
13430  050674  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13431  050676  012700  000676                  MOV     #676,R0         ;;LOAD R0 WITH TEST NUMBER
13432  050702  013701  050736                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13433  050706  013704  000014                  MOV     @#14,R4         ;SAVE BREAK POINT TRAP VECTOR
13434  050712  010605                          MOV     SP,R5           ;SAVE SP
13435  050714  012737  050752  000004          MOV     #4$,@#4 ;GO TO 4$ ON OVFLW TRAP
13436  050722  012737  050756  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13437  050730  012706  000400                  MOV     #400,SP         ;SET UP SP TO CAUSE OVFLW ON 1ST PUSH
13438  050734  000257                          CCC                     ;SCOPE SYNC
13439
13440  050736  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13441
13442  050740  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13443  050744  010506                          MOV     R5,SP           ;RESET SP
13444  050746  104005          3$:     ERROR   5                       ;BPT FAILED TO TRAP
13445
13446  050750  000406                          BR      8$              ;GO TO SCOPE EXIT
13447
13448  050752  010506                  4$:     MOV     R5,SP           ;RESET SP
13449  050754  000404                          BR      8$              ;GO EXIT - ALL OK
13450
13451  050756  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR ERROR PRINTOUT
13452  050762  010506                          MOV     R5,SP           ;RESET SP
13453  050764  104005          5$:     ERROR   5                       ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13454
13455  050766  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13456  050774  010437  000014                  MOV     R4,@#14
13457
13458                                  ;;**********************************************************************
13459                                  ;*TEST 677       TEST THAT 2ND PUSH IN TRAP MICROUTINE CAUSES OVFLW TRAP
13460                                  ;;**********************************************************************
13461  051000                          TST677:
```

```
13462  051000  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13463  051002  012700  000677                  MOV     #677,R0         ;;LOAD R0 WITH TEST NUMBER
13464  051006  013701  051042                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13465  051012  013704  000014                  MOV     @#14,R4         ;SAVE BPT VECTOR
13466  051016  010605                          MOV     SP,R5           ;SAVE SP
13467  051020  012737  051056  000004          MOV     #4$,@#4 ;GO TO 4$ ON STACK OVFLOW
13468  051026  012737  051062  000014          MOV     #6$,@#14        ;GO TO 6$ IF BPT SERVICED
13469  051034  012706  000402                  MOV     #402,SP         ;SET SP TO CAUSE TRAP ON 2ND PUSH
13470  051040  000257                          CCC                     ;SCOPE SYNC
13471
13472  051042  000003                  2$:     BPT                     ;TEST THE BPT - SHOULD CAUSE OVERFLOW TRAP
13473
13474  051044  010637  001074                  MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13475  051050  010506                          MOV     R5,SP           ;RESET SP
13476  051052  104005                  3$:     ERROR   5               ;BPT FAILED TO TRAP
13477
13478  051054  000406                          BR      8$              ;GO TO SCOPE EXIT
13479
13480  051056  010506                  4$:     MOV     R5,SP           ;RESET SP
13481  051060  000404                          BR      8$              ;GO EXIT - ALL OK
13482
13483  051062  010637  001074          6$:     MOV     SP,@#$REG5      ;SAVE BAD SP FOR PRINTING
13484  051066  010506                          MOV     R5,SP           ;RESET SP
13485  051070  104005                  5$:     ERROR   5               ;OVFLW TRAP FAILED TO BUMP BPT SERVICE
13486
13487  051072  012737  061220  000004  8$:     MOV     #BERR,@#4       ;RESET VECTORS
13488  051100  010437  000014                  MOV     R4,@#14
13489
13490                                  ;;*******************************************************************
13491                                  ;*TEST 700       ILLEGAL INSTRUCTION TEST - JSR RN,%R
13492                                  ;;*******************************************************************
13493  051104                          TST700:
13494  051104  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13495  051106  012700  000700                  MOV     #700,R0         ;;LOAD R0 WITH TEST NUMBER
13496  051112  013701  051146                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13497  051116  010605                          MOV     SP,R5           ;SAVE SP
13498  051120  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13499  051124  013704  000004          1$:     MOV     @#4,R4          ;SAVE T.O. VECTOR
13500  051130  012737  051156  000004          MOV     #4$,@#4 ;ILLEGAL INSTR. TRAP GOES TO 4$
13501  051136  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13502  051140  012702  051154                  MOV     #3$,R2          ;IN CASE JSR JUMPS TO [R2]
13503  051144  000257                          CCC                     ;SCOPE SYNC
13504
13505  051146  004302                  2$:     JSR     R3,R2           ;JSR MODE 0 FORCES TRAP - GO TO 4$
13506
13507  051150  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13508  051154  104005                  3$:     ERROR   5               ;JSR FAILED TO SPRING TRAP
13509
13510  051156  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR
13511  051162  010506                          MOV     R5,SP           ;RESET SP
13512
13513                                  ;;*******************************************************************
13514                                  ;*TEST 701       ILLEGAL INSTRUCTION TEST - JMP %R
13515                                  ;;*******************************************************************
13516  051164                          TST701:
13517  051164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
13518  051166  012700  000701                  MOV     #701,R0         ;;LOAD R0 WITH TEST NUMBER
13519  051172  013701  051226                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13520  051176  010605                          MOV     SP,R5           ;SAVE SP
13521  051200  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13522  051204  013704  000004          1$:     MOV     @#4,R4          ;SAVE VECTOR POINTER AT LOC. 4
13523  051210  012737  051236  000004          MOV     #4$,@#4 ;ON TRAP - GO TO 4$
13524  051216  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13525  051220  012702  051234                  MOV     #3$,R2          ;IN CASE IT JUMPS TO ADDR IN RN
13526  051224  000257                          CCC                     ;SCOPE SYNC
13527
13528  051226  000102                  2$:     JMP     R2              ;JMP MODE 0 FORCES TRAP - GO TO 4$
13529
13530  051230  010437  000004                  MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
13531  051234  104005                  3$:     ERROR   5               ;ILLEGAL INSTR TRAP FAILED
13532
13533  051236  010437  000004          4$:     MOV     R4,@#4          ;RESTORE VECTOR POINTER AT LOC. 4
13534  051242  010506                          MOV     R5,SP           ;RESET SP
13535
13536                                  ;;*****************************************************************
13537                                  ;*TEST 702       BUS TIMEOUT TRAP TEST - TST (R)
13538                                  ;;*****************************************************************
13539  051244                          TST702:
13540  051244  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13541  051246  012700  000702                  MOV     #702,R0         ;;LOAD R0 WITH TEST NUMBER
13542  051252  013701  051306                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13543  051256  010605                          MOV     SP,R5           ;SAVE SP
13544  051260  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13545  051264  013704  000004          1$:     MOV     @#4,R4          ;SAVE ORIGINAL T.O. VECTOR POINTER
13546  051270  012737  051316  000004          MOV     #4$,@#4 ;ON T.O. TRAP - GO TO 4$
13547  051276  012702  160000                  MOV     #160000,R2      ;ADDRESS CAUSES T.O.
13548  051302  010506                          MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13549  051304  000257                          CCC                     ;SCOPE SYNC
13550
13551  051306  005712                  2$:     TST     (R2)            ;FORCE T.O. TRAP - GO TO 4$
13552
13553  051310  010437  000004                  MOV     R4,@#4          ;RESTORE T.O. VECTOR
13554  051314  104005                  3$:     ERROR   5               ;TIMEOUT TRAP FAILED
13555  051316  010437  000004          4$:     MOV     R4,@#4          ;RESTORE T.O. VECTOR
13556  051322  010506                          MOV     R5,SP           ;RESET SP
13557
13558                                  ;;*****************************************************************
13559                                  ;*TEST 703       "T" BIT TRAP TEST
13560                                  ;;*****************************************************************
13561  051324                          TST703:
13562  051324  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13563  051326  012700  000703                  MOV     #703,R0         ;;LOAD R0 WITH TEST NUMBER
13564  051332  013701  051370                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13565  051336  010605                          MOV     SP,R5           ;SAVE SP
13566  051340  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13567  051344  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP
13568  051346  012737  051376  000014          MOV     #4$,@#14        ;GO TO 4$ WHEN "T" TRAP SPRUNG
13569  051354  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT ON STACK
13570  051360  012746  051370                  MOV     #2$,-(SP)       ;SET UP NEW PC ON STACK
13571  051364  000257                          CCC                     ;SCOPE SYNC
13572  051366  000006                          RTT                     ;TURN ON "T" BIT - GO TO 2$
13573
```

```
13574  051370  005700                  2$:     TST     R0              ;SPRING "T" BIT TRAP - GO TO 4$
13575
13576  051372  104005                  3$:     ERROR   5               ;NO "T" BIT TRAP OCCURRED
13577
13578  051374  000405                          BR      6$              ;GO EXIT
13579
13580  051376  032766  000020  000002  4$:     BIT     #20,2(SP)       ;"T" BIT SET IN OLD PSW?
13581  051404  001001                          BNE     6$              ;BR IF YES
13582
13583  051406  104001                  5$:     ERROR   1               ;#T# BIT NOT SAVED ON STACK
13584
13585  051410  012737  000016  000014  6$:     MOV     #16,@#14        ;RESTORE "T" BIT TRAP CATCHER
13586  051416  005037  000016                  CLR     @#16
13587  051422  010506                          MOV     R5,SP           ;RESET SP
13588
13589                                  ;;*********************************************************
13590                                  ;*TEST 704      TEST PUSH INTO PSW WITH [SP] = 000000
13591                                  ;THESE NEXT TWO TESTS VERIFY THAT A "RED ZONE" TRAP OCCURS IF A
13592                                  ;PUSH IS ATTEMPTED WITH THE [SP] INITIALLY EQUAL TO 000000,177572.
13593                                  ;;*********************************************************
13594  051424                          TST704:
13595  051424  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13596  051426  012700  000704                  MOV     #704,R0         ;;LOAD R0 WITH TEST NUMBER
13597  051432  013701  051456                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13598  051436  010605                          MOV     SP,R5           ;SAVE THE SP
13599  051440  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13600  051444  012737  051500  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13601  051452  005006                          CLR     SP              ;MAKE SP - 000000
13602  051454  000257                          CCC                     ;SCOPE SYNC
13603
13604  051456  012746  007777          2$:     MOV     #7777,-(SP)     ;ATTEMPT PUSH INTO PSW - SHOULD CAUSE
13605                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
13606
13607  051462  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13608  051466  005004                          CLR     R4              ;[R4] = S / B SP
13609  051470  010603                          MOV     SP,R3           ;[R3] = WAS SP
13610  051472  010506                          MOV     R5,SP           ;RESET THE SP
13611  051474  104003                  3$:     ERROR   3               ;TRAP NOT SPRUNG
13612  051476  000414                          BR      TST705          ;;GO TO SCOPE EXIT - SCHOOL'S OU
13613
13614  051500  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13615  051504  001406                          BEQ     6$              ;BR IF YES
13616
13617  051506  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13618  051512  005004                          CLR     R4              ;[R4]= S / B SP
13619  051514  010603                          MOV     SP,R3           ;[R3] = WAS SP
13620  051516  010506                          MOV     R5,SP           ;RESET THE SP
13621  051520  104003                  5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13622
13623  051522  010506                  6$:     MOV     R5,SP           ;FIX UP THE SP
13624  051524  010437  000004                  MOV     R4,@#4          ;RESTORE BERR VECTOR
13625
13626                                  ;;*********************************************************
13627                                  ;*TEST 705      TEST PUSH INTO SR WITH [SP] = 177572
13628                                  ;;*********************************************************
13629  051530                          TST705:
```

```
13630  051530  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13631  051532  012700  000705                  MOV     #705,R0         ;;LOAD R0 WITH TEST NUMBER
13632  051536  013701  051564                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13633  051542  010605                          MOV     SP,R5           ;SAVE THE SP
13634  051544  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13635  051550  012737  051606  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13636  051556  012706  177572                  MOV     #177572,SP      ;MAKE SP=177572
13637  051562  000257                          CCC                     ;SCOPE SYNC
13638
13639  051564  012746  177777          2$:     MOV     #-1,-(SP)       ;ATTEMPT PUSH INTO SR - SHOULD CAUSE
13640                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
13641
13642  051570  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13643  051574  005004                          CLR     R4              ;[R4] = S / B SP
13644  051576  010603                          MOV     SP,R3           ;[R3] = WAS SP
13645  051600  010506                          MOV     R5,SP           ;RESET THE SP
13646  051602  104003          3$:     ERROR   3                       ;TRAP NOT SPRUNG
13647  051604  000414                          BR      TST706          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13648
13649  051606  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13650  051612  001406                          BEQ     6$              ;BR IF YES
13651
13652  051614  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13653  051620  005004                          CLR     R4              ;[R4]= S / B SP
13654  051622  010603                          MOV     SP,R3           ;[R3] = WAS SP
13655  051624  010506                          MOV     R5,SP           ;RESET THE SP
13656  051626  104003          5$:     ERROR   3                       ;TRAP SPRUNG BUT NOT RED ZONE
13657
13658  051630  010506          6$:     MOV     R5,SP           ;FIX UP THE SP
13659  051632  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13660
13661                          ;;*************************************************************
13662                          ;*TEST 706       TEST PUSH INTO SLR WITH [SP] = 177776
13663                          ;;*************************************************************
13664  051636                  TST706:
13665  051636  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13666  051640  012700  000706                  MOV     #706,R0         ;;LOAD R0 WITH TEST NUMBER
13667  051644  013701  051672                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13668  051650  010605                          MOV     SP,R5           ;SAVE THE SP
13669  051652  013704  000004                  MOV     @#4,R4          ;SAVE THE BUS ERROR VECTOR
13670  051656  012737  051714  000004          MOV     #4$,@#4 ;"RED ZONE" TRAP GOES TO 4$
13671  051664  012706  177776                  MOV     #177776,SP      ;MAKE SP=177776
13672  051670  000257                          CCC                     ;SCOPE SYNC
13673
13674  051672  012746  000200          2$:     MOV     #200,-(SP)      ;ATTEMPT PUSH INTO SLR - SHOULD CAUSE
13675                                                                  ;"RED ZONE" TRAP TO BE SPRUNG
13676
13677  051676  010437  000004                  MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13678  051702  005004                          CLR     R4              ;[R4] = S / B SP
13679  051704  010603                          MOV     SP,R3           ;[R3] = WAS SP
13680  051706  010506                          MOV     R5,SP           ;RESET THE SP
13681  051710  104003          3$:     ERROR   3                       ;TRAP NOT SPRUNG
13682  051712  000414                          BR      TST707          ;;GO TO SCOPE EXIT - SCHOOL'S OUT
13683
13684  051714  022706  000000          4$:     CMP     #0,SP           ;WAS IT A RED ZONE TRAP ?
13685  051720  001406                          BEQ     6$              ;BR IF YES
```

```
13686
13687  051722  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13688  051726  005004                  CLR     R4              ;[R4]= S / B SP
13689  051730  010603                  MOV     SP,R3           ;[R3] = WAS SP
13690  051732  010506                  MOV     R5,SP           ;RESET THE SP
13691  051734  104003          5$:     ERROR   3               ;TRAP SPRUNG BUT NOT RED ZONE
13692
13693  051736  010506          6$:     MOV     R5,SP           ;FIX UP THE SP
13694  051740  010437  000004          MOV     R4,@#4          ;RESTORE BUS ERROR VECTOR
13695
13696                          ;;*****************************************************************
13697                          ;*TEST 707      RSVD INSTRUCTION TEST - 000007 THRU 000077
13698                          ;;*****************************************************************
13699  051744                  TST707:
13700  051744  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13701  051746  012700  000707          MOV     #707,R0         ;;LOAD R0 WITH TEST NUMBER
13702  051752  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13703  051754  012737  052012  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13704  051762  005037  000012          CLR     @#12
13705  051766  012701  000007          MOV     #7,R1           ;SET UP FIRST ONE IN GROUP
13706  051772  010737  001010          MOV     PC,@#$LPERR     ;ONLY LOOP ON BAD OP CODE
13707  051776  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13708  052000  010137  052006          MOV     R1,@#2$ ;LOAD NEW INSTR
13709  052004  000257                  CCC                     ;SCOPE SYNC
13710
13711  052006  000007          2$:     000007                  ;TEST THE RSVD INSTR - THIS LOCATION
13712                                                          ;GETS CHANGED EACH PASS THROUGH
13713
13714  052010  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13715
13716  052012  005201          4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13717  052014  022701  000100          CMP     #100,R1         ;AT END OF THIS GROUP ??
13718  052020  001366                  BNE     1$              ;BR IF NOT
13719
13720  052022  010506                  MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13721  052024  012737  051752  001010  MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13722                          ;;*****************************************************************
13723                          ;*TEST 710      RSVD INSTRUCTION TEST - 000210 THRU 000237
13724                          ;;*****************************************************************
13725  052032                  TST710:
13726  052032  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13727  052034  012700  000710          MOV     #710,R0         ;;LOAD R0 WITH TEST NUMBER
13728  052040  010605          5$:     MOV     SP,R5           ;SAVE THE SP
13729  052042  012737  052100  000010  MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13730  052050  005037  000012          CLR     @#12
13731  052054  012701  000210          MOV     #210,R1         ;SET UP FIRST ONE IN GROUP
13732  052060  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13733  052064  010506          1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13734  052066  010137  052074          MOV     R1,@#2$ ;LOAD NEW INSTR
13735  052072  000257                  CCC                     ;SCOPE SYNC
13736
13737  052074  000210          2$:     000210                  ;TEST THE RSVD INSTR - THIS LOCATION
13738                                                          ;GETS CHANGED EACH PASS THROUGH
13739
13740  052076  104005          3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13741
```

```
13742  052100  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13743  052102  022701  000240                  CMP     #240,R1         ;AT END OF THIS GROUP ??
13744  052106  001366                          BNE     1$              ;BR IF NOT
13745
13746  052110  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13747  052112  012737  052040  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13748
13749                                  ;;*****************************************************************
13750                                  ;*TEST 711       RSVD INSTRUCTION TEST - 007000 THRU 007777
13751                                  ;;*****************************************************************
13752  052120                          1ST711:
13753  052120  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13754  052122  012700  000711                  MOV     #711,R0         ;;LOAD R0 WITH TEST NUMBER
13755  052126  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13756  052130  012737  052166  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13757  052136  005037  000012                  CLR     @#12
13758  052142  012701  007000                  MOV     #7000,R1        ;SET UP FIRST ONE IN GROUP
13759  052146  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13760  052152  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13761  052154  010137  052162                  MOV     R1,@#2$ ;LOAD NEW INSTR
13762  052160  000257                          CCC                     ;SCOPE SYNC
13763
13764  052162  007000                  2$:     007000                  ;TEST THE RSVD INSTR - THIS LOCATION
13765                                                                  ;GETS CHANGED EACH PASS THROUGH
13766
13767  052164  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13768
13769  052166  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13770  052170  022701  010000                  CMP     #10000,R1       ;AT END OF THIS GROUP ??
13771  052174  001366                          BNE     1$              ;BR IF NOT
13772
13773  052176  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13774  052200  012737  052126  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13775
13776                                  ;;*****************************************************************
13777                                  ;*TEST 712       RSVD INSTRUCTION TEST - 075000 THRU 076777
13778                                  ;;*****************************************************************
13779  052206                          TST712:
13780  052206  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13781  052210  012700  000712                  MOV     #712,R0         ;;LOAD R0 WITH TEST NUMBER
13782  052214  010605                  5$:     MOV     SP,R5           ;SAVE THE SP
13783  052216  012737  052256  000010          MOV     #4$,@#10        ;SET UP RSVD INSTR. TRAP VECTOR
13784  052224  005037  000012                  CLR     @#12
13785  052230  012701  075000                  MOV     #75000,R1       ;SET UP FIRST ONE IN GROUP
13786  052234  010737  001010                  MOV     PC,@#$LPERR     ;SET ERROR LOOP ADDRESS
13787  052240  010506                  1$:     MOV     R5,SP           ;RESET SP FOR ERROR LOOP AND NEW INSTR
13788  052242  010137  052250                  MOV     R1,@#2$ ;LOAD NEW INSTR
13789  052246  000257                          CCC                     ;SCOPE SYNC
13790
13791  052250  075000                  2$:     75000                   ;TEST THE RSVD INSTR - THIS LOCATION
13792                                                                  ;GETS CHANGED EACH PASS THROUGH
13793
13794  052252  000240                          NOP                     ;IN CASE NON TRAPPING INSTR IS TWO WORDS
13795  052254  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13796
13797  052256  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
```

```
13798  052260  022701  076600                 CMP     #MED,R1          ;MED INSTRUCTION?
13799  052264  001774                         BEQ     4$               ;BR IF YES--SKIP IT.
13800  052266  022701  077000                 CMP     #077000,R1       ;AT END OF THIS GROUP ??
13801  052272  001362                         BNE     1$               ;BR IF NOT
13802
13803  052274  010506                         MOV     R5,SP            ;MAKE SURE TO RESET THE SP
13804  052276  012737  052214  001010         MOV     #5$,@#$LPERR     ;LOOP FROM BEGINNING ON ERROR
13805
13806                                 ;;*****************************************************************
13807                                 ;*TEST 713      RSVD INSTRUCTION TEST - 106400 THRU 107777
13808                                 ;;*****************************************************************
13809  052304                         TST713:
13810  052304  000004                         SCOPE                    ;CALL THE SCOPE LOOP UTILITY
13811  052306  012700  000713                 MOV     #713,R0          ;;LOAD R0 WITH TEST NUMBER
13812  052312  010605                 5$:     MOV     SP,R5            ;SAVE THE SP
13813  052314  012737  052352  000010         MOV     #4$,@#10         ;SET UP RSVD INSTR. TRAP VECTOR
13814  052322  005037  000012                 CLR     @#12
13815  052326  012701  106400                 MOV     #106400,R1       ;SET UP FIRST ONE IN GROUP
13816  052332  010737  001010                 MOV     PC,@#$LPERR      ;SET ERROR LOOP ADDRESS
13817  052336  010506                 1$:     MOV     R5,SP            ;RESET SP FOR ERROR LOOP AND NEW INSTR
13818  052340  010137  052346                 MOV     R1,@#2$ ;LOAD NEW INSTR
13819  052344  000257                         CCC                      ;SCOPE SYNC
```

```
13820
13821  052346  106400                  2$:     106400                  ;TEST THE RSVD INSTR - THIS LOCATION
13822                                                                   ;GETS CHANGED EACH PASS THROUGH
13823
13824  052350  104005                  3$:     ERROR   5               ;RSVD INSTR. IN R1 FAILED TO TRAP
13825
13826  052352  005201                  4$:     INC     R1              ;GENERATE NEW RSVD INSTR
13827  052354  022701  106500                  CMP     #106500,R1      ;MFPD INSTRUCTION ??
13828  052360  001002                          BNE     10$             ;BR IF NOT
13829  052362  012701  106700                  MOV     #106700,R1      ;SKIP MFPD AND MTPD INSTRUCTIONS
13830  052366  022701  110000          10$:    CMP     #110000,R1      ;AT END OF THIS GROUP ??
13831  052372  001361                          BNE     1$              ;BR IF NOT
13832
13833  052374  010506                          MOV     R5,SP           ;MAKE SURE TO RESET THE SP
13834  052376  012737  052312  001010          MOV     #5$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13835  052404  012737  061122  000010          MOV     #RSERR,@#10     ;RESTORE RSVD INSTR VECTOR
13836  052412  012737  000340  000012          MOV     #340,@#12
13837  052420  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13838
13839                                  ;THIS NEXT GROUP OF SEQUENTIAL TESTS VERIFIES THAT A "T" BIT
13840                                  ;TRAP CAN BE SERVICED IN EACH MICROWORD THAT DOES A "BUT SERVICE"
13841                                  ;EACH ROUTINE ENTERS THE TRAP MICROUTINE WHEN THE TRAP IS SPRUNG
13842
13843  052422  012737  061070  000014  TSET:   MOV     #TBSER,@#14     ;SET UP THE "T" BIT TRAP VECTOR
13844  052430  012737  000340  000016          MOV     #340,@#16       ;PRIORITY 7
13845
13846                                  ;;*****************************************************************
13847                                  ;*TEST 714      BUT SERVICE -- ONE WORD INSTRUCTIONS-ALL MODES -- FROM TABLE
13848                                  ;"INSTAB" (INSTRUCTION TABLE) CONTAINS ALL ONE WORD INSTRUCTIONS
13849                                  ;THAT TEST A "BUT SERVICE" IN A UNIQUE ROM LOCATION.  THE TABLE MUST
13850                                  ;BE TERMINATED WITH A 0 ENTRY.
13851                                  ;;*****************************************************************
13852  052436                          TST714:
13853  052436  012700  000714                  MOV     #714,R0         ;;LOAD R0 WITH TEST NUMBER
13854  052442  010605                  6$:     MOV     SP,R5           ;SAVE THE SP
13855  052444  012704  063636                  MOV     #INSTAB,R4      ;PUT POINTER TO TABLE IN R4
13856  052450  012401                  4$:     MOV     (R4)+,R1        ;LOAD R1 WITH TEST INSTRUCTION WORD
13857  052452  001422                          BEQ     5$              ;EXIT TEST IF END OF TABLE
13858  052454  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ON FAILING INSTRUCTION ONLY
13859  052460  010137  052512          1$:     MOV     R1,@#2$         ;STORE TEST INSTRUCTION TO BE EXECUTED
13860  052464  012702  063312                  MOV     #MBUF0,R2       ;IN CASE DM1 DEST--(R2)
13861  052470  012703  063316                  MOV     #MBUF1,R3       ;IN CASE SM1--(R3)
13862  052474  010506                          MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13863  052476  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13864  052502  012746  052512                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13865  052506  000257                          CCC                     ;SCOPE SYNC
13866  052510  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13867
13868  052512  000240          2$:     NOP                             ;INSTRUCTION FROM TABLE IS STORED HERE AND
13869                                                                   ;SHOULD SPRING TRAP
13870
13871  052514  104005                  3$:     ERROR   5               ;BUT SERVICE FAILED
13872
13873  052516  000754                          BR      4$              ;GET NEXT INSTRUCTION FOR BUT SERVICE TEST
13874  052520  012737  052442  001010  5$:     MOV     #6$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
13875
```

```
13876                                   ;;*****************************************************************
13877                                   ;*TEST 715      BUT SERVICE TEST - (RTI)
13878                                   ;;*****************************************************************
13879  052526                           TST715:
13880  052526  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13881  052530  012700  000715                  MOV     #715,R0         ;;LOAD R0 WITH TEST NUMBER
13882  052534  013701  052552                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13883  052540  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13884  052544  012746  052554                  MOV     #3$,-(SP)       ;MAKE NEW PC = 3$
13885  052550  000257                          CCC                     ;SCOPE SYNC
13886
13887  052552  000002                  2$:     RTI                     ;INSTRUCTION SHOULD SPRING TRAP
13888
13889  052554  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13890
13891                                   ;;*****************************************************************
13892                                   ;*TEST 716      BUT SERVICE TEST - (JSR  %R,@A)
13893                                   ;;*****************************************************************
13894  052556                           TST716:
13895  052556  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13896  052560  012700  000716                  MOV     #716,R0         ;;LOAD R0 WITH TEST NUMBER
13897  052564  013701  052634                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13898                                   .SBTTL USER CONTROLLED BREAKPOINT -- BIT14
13899  052570  032737  040000  063234          BIT     #BIT14,@#BPTLOC ;BREAKPOINT HALT SET ??
13900  052576  001401                          BEQ     .+4             ;BR IF NOT
13901  052600  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
13902  052602  010605                          MOV     SP,R5           ;SAVE THE SP
13903  052604  010737  001010                  MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13904  052610  010506                  1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13905  052612  012737  052640  063316          MOV     #3$,@#MBUF1     ;SET UP POINTER--DEST ADDR = 3$ FOR JSR
13906  052620  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13907  052624  012746  052634                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13908  052630  000257                          CCC                     ;SCOPE SYNC
13909  052632  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13910
13911  052634  004777  010456          2$:     JSR     PC,@MBUF1       ;INSTRUCTION SHOULD SPRING TRAP
13912
13913  052640  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13914
13915  052642  010506                          MOV     R5,SP           ;RESTORE SP IF ALL OK OR NOT LOOPING
13916                                   ;;*****************************************************************
13917                                   ;*TEST 717      BUT SERVICE TEST - (JMP A)
13918                                   ;;*****************************************************************
13919  052644                           TST717:
13920  052644  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13921  052646  012700  000717                  MOV     #717,R0         ;;LOAD R0 WITH TEST NUMBER
13922  052652  013701  052672                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13923  052656  012746  000020                  MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13924  052662  012746  052672                  MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13925  052666  000257                          CCC                     ;SCOPE SYNC
13926  052670  000006                          RTT                     ;SET "T" BIT - GO TO 2$
13927
13928  052672  000167  000000          2$:     JMP     3$              ;JMP INSTRUCTION SHOULD SPRING TRAP
13929
13930  052676  104005                  3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13931
```

```
13932                                   ;;*****************************************************************
13933                                   ;*TEST 720      BUT SERVICE TEST - (JMP @A)
13934                                   ;;*****************************************************************
13935  052700                           TST720:
13936  052700  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13937  052702  012700  000720                   MOV     #720,R0         ;;LOAD R0 WITH TEST NUMBER
13938  052706  013701  052734                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13939  052712  012737  052740  063312           MOV     #3$,@#MBUF0     ;SET UP POINTER--DEST ADDR = 3$ FOR JMP
13940  052720  012746  000020                   MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13941  052724  012746  052734                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13942  052730  000257                           CCC                     ;SCOPE SYNC
13943  052732  000006                           RTT                     ;SET "T" BIT - GO TO 2$
13944
13945  052734  000177  010352           2$:     JMP     @MBUF0          ;JMP INSTRUCTION SHOULD SPRING TRAP
13946
13947  052740  104005                   3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13948
13949                                   ;;*****************************************************************
13950                                   ;*TEST 721      BUT SERVICE TEST - (RTS PC)
13951                                   ;;*****************************************************************
13952  052742                           TST721:
13953  052742  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
13954  052744  012700  000721                   MOV     #721,R0         ;;LOAD R0 WITH TEST NUMBER
13955  052750  013701  053004                   MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
13956  052754  010605                           MOV     SP,R5           ;SAVE THE SP
13957  052756  010737  001010                   MOV     PC,@#$LPERR     ;FOR PROPER SP RESETTING ON ERROR LOOP
13958  052762  010506           1$:     MOV     R5,SP           ;RESTORE SP FOR ERROR LOOPING
13959  052764  012746  053006                   MOV     #3$,-(SP)       ;RTS WILL LOAD PC WITH 3$
13960  052770  012746  000020                   MOV     #20,-(SP)       ;SET "T" BIT IN THE NEW PSW
13961  052774  012746  053004                   MOV     #2$,-(SP)       ;MAKE NEW PC = 2$
13962  053000  000257                           CCC                     ;SCOPE SYNC
13963  053002  000006                           RTT                     ;SET "T" BIT - GO TO 2$
13964
13965  053004  000207           2$:     RTS     PC              ;RTS INSTRUCTION SHOULD SPRING TRAP
13966
13967  053006  104005           3$:     ERROR   5               ;BUT SERVICE IN XXX FAILED
13968
13969                                   ;;*****************************************************************
13970                                   ;*TEST 722      ALU ADD FUNCTION TEST
13971                                   ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
13972                                   ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
13973                                   ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
13974
13975                                   ;       AIN     BIN     CIN
13976
13977                                   ;       0       0       0
13978                                   ;       0       0       1
13979                                   ;       0       1       0
13980                                   ;       0       1       1
13981                                   ;       1       0       0
13982                                   ;       1       0       1
13983                                   ;       1       1       0
13984                                   ;       1       1       1
13985
13986                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
13987                                   ;TAGGED "ALUADD" AS SHOWN BELOW:
```

```
13988
13989                                   ;ALUADD:        NULL
13990                                   ;               SRC OP1
13991                                   ;               DST OP1
13992                                   ;               SUM1
13993                                   ;               SRC OP2
13994                                   ;               DST OP2
13995                                   ;               SUM2
13996                                   ;               ETC.
13997
13998                                   ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
13999                                   ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14000                                   ;;*****************************************************************
14001  053010                           TST722:
14002  053010  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14003  053012  012700  000722                  MOV     #722,R0         ;;LOAD R0 WITH TEST NUMBER
14004  053016  012705  063340          1$:     MOV     #ALUADD+4,R5    ;R5 POINTS TO TABLE OF NO.S
14005  053022  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14006  053026  024545                          CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14007                                                                   ;(OR NULL ENTRY FIRST TIME THROUGH)
14008  053030  005725                  4$:     TST     (R5)+           ;POINT TO A SRC OP
14009  053032  022705  063416                  CMP     #ALUADD+62,R5   ;DONE ALL NO.S IN TABLE ?
14010  053036  001413                          BEQ     5$              ;BR IF YES
14011  053040  012501                          MOV     (R5)+,R1        ;LOAD SRC OP
14012  053042  012503                          MOV     (R5)+,R3        ;LOAD DEST OP
14013  053044  000257                          CCC                     ;SCOPE SYNC
14014
14015  053046  060103                  2$:     ADD     R1,R3           ;TEST THE ADD FUNCTION
14016
14017  053050  021503                          CMP     (R5),R3         ;CORRECT SUM ?
14018  053052  001766                          BEQ     4$              ;GO ADD NEXT PAIR IF YES
14019
14020  053054  011504                          MOV     (R5),R4         ;GET S / B SUM
14021  053056  014502                          MOV     -(R5),R2        ;GET DEST OP
14022  053060  104010                  3$:     ERROR   10              ;ALU ADD OPERATION FAILED
14023
14024  053062  005725                          TST     (R5)+           ;CORRECT R5 POINTER
14025  053064  000761                          BR      4$              ;GO DO NEXT PAIR
14026
14027  053066  012737  053016  001010  5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
14028
14029                                   ;;*****************************************************************
14030                                   ;*TEST 723       ALU SUB FUNCTION TEST
14031                                   ;THIS TEST VERIFIES THAT THE ALU ADD FUNCTION CAN RESPOND CORRECTLY
14032                                   ;TO THE 8 POSSIBLE COMBINATIONS THAT COULD OCCUR AT THE INPUTS OF
14033                                   ;EACH OF THE 16 BIT POSITIONS AS DESCRIBED BELOW:
14034
14035                                   ;       AIN     BIN     CIN
14036
14037                                   ;       0       0       0
14038                                   ;       0       0       1
14039                                   ;       0       1       0
14040                                   ;       0       1       1
14041                                   ;       1       0       0
14042                                   ;       1       0       1
14043                                   ;       1       1       0
```

```
14044                                   ;       1       1       1
14045
14046                                   ;THE TEST NO.S ALONG WITH THE CORRECT ANSWERS ARE STORED IN A TABLE
14047                                   ;TAGGED "ALUADD" AS SHOWN BELOW:
14048
14049                                           ;ALUSUB:        NULL
14050                                   ;                       SRC OP1
14051                                   ;                       DST OP1
14052                                   ;                       DIFF1
14053                                   ;                       SRC OP2
14054                                   ;                       DST OP2
14055                                   ;                       DIFF2
14056                                   ;                       ETC.
14057
14058                                   ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING PAIR
14059                                   ;OF NO.S IF SW09=1 OR GO ON TO THE NEXT PAIR IF SW09=0.
14060                                   ;;*****************************************************************
14061  053074                           TST723:
14062  053074  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14063  053076  012700  000723                   MOV     #723,R0         ;;LOAD R0 WITH TEST NUMBER
14064  053102  012705  063560           1$:     MOV     #ALUSUB+4,R5    ;R5 POINTS TO TABLE OF NO.S
14065  053106  010737  001010                   MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14066  053112  024545                           CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
14067                                                                   ;(OR NULL ENTRY FIRST TIME THROUGH)
14068  053114  005725                   4$:     TST     (R5)+           ;POINT TO A SRC OP
14069  053116  022705  063636                   CMP     #ALUSUB+62,R5   ;DONE ALL NO.S IN TABLE ?
14070  053122  001413                           BEQ     5$              ;BR IF YES
14071  053124  012501                           MOV     (R5)+,R1        ;LOAD SRC OP
14072  053126  012503                           MOV     (R5)+,R3        ;LOAD DEST OP
14073  053130  000257                           CCC                     ;SCOPE SYNC
14074
14075  053132  160103                   2$:     SUB     R1,R3           ;TEST THE SUB FUNCTION
14076
14077  053134  021503                           CMP     (R5),R3         ;CORRECT DIFF. ?
14078  053136  001766                           BEQ     4$              ;GO SUB NEXT PAIR IF YES
14079
14080  053140  011504                           MOV     (R5),R4         ;GET S / B DIFF
14081  053142  014502                           MOV     -(R5),R2        ;GET DEST OP
14082  053144  104010                   3$:     ERROR   10              ;ALU SUB OPERATION FAILED
14083
14084  053146  005725                           TST     (R5)+           ;CORRECT R5 POINTER
14085  053150  000761                           BR      4$              ;GO DO NEXT PAIR
14086
14087  053152  012737  053102  001010   5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
14088
14089                                   ;;*****************************************************************
14090                                   ;*TEST 724      ALU "AND" FUNCTION TEST USING BIC INSTRUCTION
14091                                   ;THIS TEST VERIFIES THAT THE ALU "AND" FUNCTION RESPONDS CORRECTLY
14092                                   ;TO ALL POSSIBLE COMBINATIONS FOR EACH OF THE 16 BIT POSITIONS
14093                                   ;IT  EXECUTES THE BIC INSTRUCTION FOR THE FOLLOWING PAIRS OF
14094                                   ;OPERANDS AND TESTS FOR THE INDICATED RESULT:
14095
14096                                           ;SOURCE OP      DEST. OP        RESULT
14097
14098                                           ;000000         000000          000000
14099                                           ;177777         177777          000000
```

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

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

                                ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
                                ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
                                ;;*****************************************************************
053160                          TST724:
                                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
053160  000004                  1$:     MOV     #724,R0         ;;LOAD R0 WITH TEST NUMBER
053162  012700  000724                  MOV     #ANDTAB+4,R5    ;R5 POINTS TO TABLE OF TEST NO.S
053166  012705  063420                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
053172  010737  001010                  CMP     -(R5),-(R5)     ;RESET R5 TO POINT TO BAD GUYS
053176  024545                                                  ;(OR NULL ENTRY FIRST TIME THROUGH)
                                4$:     TST     (R5)+           ;POINT TO A SOURCE OPR
053200  005725                          CMP     #ANDTAB+62,R5   ;DONE ALL COMBINATIONS ?
053202  022705  063476                  BEQ     5$              ;BR IF
053206  001413                          MOV     (R5)+,R1        ;LOAD THE SRC OP
053210  012501                          MOV     (R5)+,R3        ;LOAD THE DEST OP
053212  012503                          CCC                     ;SCOPE SYNC
053214  000257
                                2$:     BIC     R1,R3           ;TEST THE "AND"
053216  040103
                                        CMP     R3,(R5)         ;RESULT CORRECT ?
053220  020315                          BEQ     4$              ;BR IF YES - GET THE NEXT PAIR
053222  001766
                                        MOV     (R5),R4         ;GET THE S / B DATA
053224  011504                          MOV     -(R5),R2        ;GET DEST OP
053226  014502                  3$:     ERROR   10              ;ALU "AND" FAILED
053230  104010                          TST     (R5)+           ;CORRECT R5 POINTER
053232  005725                          BR      4$              ;GO GET NEXT PAIR
053234  000761
                                5$:     MOV     #1$,@#$LPERR    ;LOOP FROM BEGINNING ON ERROR
053236  012737  053166  001010

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

```
14156                                                ;SOURCE OP      DEST. OP        RESULT
14157
14158                                                ;000000         000000          000000
14159                                                ;177777         177777          177777
14160                                                ;000000         177777          177777
14161                                                ;177777         000000          177777
14162                                                ;125252         125252          125252
14163                                                ;052525         052525          052525
14164                                                ;125252         052525          177777
14165                                                ;052525         125252          177777
14166
14167                                              ;THE 8 PAIRS OF NO.S AND THE ANSWERS ARE STORED IN A TABLE TAGGED
14168                                              ;"ORTAB" IN THE FOLLOWING PATTERN:
14169                                                      ;ORTAB: NULL
14170                                              ;                       SRC OP1
14171                                              ;                       DST OP1
14172                                              ;                       ANS1
14173                                              ;                       SRC OP2
14174                                              ;                       DST OP2
14175                                              ;                       ANS2
14176                                              ;                       ETC.
14177
14178                                              ;AFTER REPORTING THE ERROR THE ROUTINE WILL LOCK ON THE FAILING
14179                                              ;PAIR OF NO.S IF SW09=1 OR GO ON TO TEST THE NEXT PAIR IF SW09=0
14180                                              ;;**************************************************************
14181  053244                                      TST725:
14182  053244  000004                                     SCOPE                           ;CALL THE SCOPE LOOP UTILITY
14183  053246  012700  000725                             MOV     #725,R0                 ;;LOAD R0 WITH TEST NUMBER
14184  053252  012705  063500                     1$:     MOV     #ORTAB+4,R5             ;R5 POINTS TO TABLE OF TEST NO.S
14185  053256  010737  001010                             MOV     PC,@#$LPERR             ;LOOP ONLY ON FAILING PAIR OF #'S
14186  053262  024545                                     CMP     -(R5),-(R5)             ;RESET R5 TO POINT TO BAD GUYS
14187                                                                                   ;(OR NULL ENTRY FIRST TIME THROUGH)
14188  053264  005725                             4$:     TST     (R5)+                   ;POINT TO A SOURCE OPR
14189  053266  022705  063556                             CMP     #ORTAB+62,R5            ;DONE ALL COMBINATIONS ?
14190  053272  001413                                     BEQ     5$                      ;BR IF YES
14191  053274  012501                                     MOV     (R5)+,R1                ;LOAD THE SRC OP
14192  053276  012503                                     MOV     (R5)+,R3                ;LOAD THE DEST OP
14193  053300  000257                                     CCC                             ;SCOPE SYNC
14194
14195  053302  050103                             2$:     BIS     R1,R3                   ;TEST THE "OR"
14196
14197  053304  020315                                     CMP     R3,(R5)                 ;RESULT CORRECT ?
14198  053306  001766                                     BEQ     4$                      ;BR IF YES - GET THE NEXT PAIR
14199
14200  053310  011504                                     MOV     (R5),R4                 ;GET THE S / B DATA
14201  053312  014502                                     MOV     -(R5),R2                ;GET DEST OP
14202  053314  104010                             3$:     ERROR   10                      ;ALU "OR" FAILED
14203
14204  053316  005725                                     TST     (R5)+                   ;CORRECT R5 POINTER
14205  053320  000761                                     BR      4$                      ;GO GET NEXT PAIR
14206
14207  053322  012737  053252  001010             5$:     MOV     #1$,@#$LPERR            ;LOOP FROM BEGINNING ON ERROR
14208
14209                                              ;;**************************************************************
14210                                              ;*TEST 726      INC / DEC / ADD TEST - CYCLE NO.S 000000-077777
14211                                              ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
```

```
14212                                   ;TEST SEQUENCE:
14213
14214                                           ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14215                                           ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14216                                           ;3. THE SOURCE OP IS INCREMENTED
14217                                           ;4. THE DEST OP IS DECREMENTED
14218                                           ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE SOURCE OP GOES
14219                                           ;   NEGATIVE
14220
14221                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14222
14223                                   ;       1. SW09=0  THE TEST IS EXITED
14224                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14225
14226                                   ;;*****************************************************************
14227  053330                           TST726:
14228  053330  000004                           SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14229  053332  012700  000726                   MOV     #726,R0         ;;LOAD R0 WITH TEST NUMBER
14230  053336  005001                   10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14231  053340  005002                           CLR     R2
14232  053342  005004                           CLR     R4
14233  053344  010737  001010                   MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14234  053350  010203                   1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14235  053352  000257                           CCC                     ;SCOPE SYNC
14236
14237  053354  060103                   2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14238                                                                   ;RESULT S / B = 000000
14239
14240  053356  020403                           CMP     R4,R3           ;RESULT = 000000 ?
14241  053360  001402                           BEQ     4$              ;BR IF YES
14242
14243  053362  104010                   3$:     ERROR   10              ;INCORRECT RESULT IN R3
14244
14245  053364  000407                           BR      TST727          ;;EXIT TO NEXT TEST
14246
14247  053366  005201                   4$:     INC     R1              ;ADD 1 TO SOURCE OP
14248  053370  100402                           BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14249  053372  005302                           DEC     R2              ;SUB 1 FROM THE DEST OP
14250  053374  000765                           BR      1$              ;GO ADD THE TWO NO.S
14251
14252  053376  012737  053336  001010   5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14253
14254                                   ;;*****************************************************************
14255                                   ;*TEST 727      INC / DEC / ADD TEST - CYCLE NO.S 077777-000000
14256                                   ;THIS TEST COMBINES THE INC / DEC / ADD INSTRUCTIONS IN THE FOLLOWING
14257                                   ;TEST SEQUENCE:
14258
14259                                           ;1. BOTH SOURCE AND DEST OPS ARE ZEROED
14260                                           ;2. THE TWO NO.S ARE ADDED AND THE RESULT COMPARED WITH 000000
14261                                           ;3. THE SOURCE OP IS DECREMENTED
14262                                           ;4. THE DEST OP IS INCREMENTED
14263                                           ;5. STEPS 2,3, AND 4 ARE REPEATED UNTIL THE DEST. OP GOES
14264                                           ;   NEGATIVE
14265
14266                                   ;ON DETECTION OF A NON-ZERO RESULT THE ERROR IS REPORTED AND THEN IF:
14267
```

```
14268                                   ;       1. SW09=0  THE TEST IS EXITED
14269                                   ;       2. SW09=1  THE ROUTINE LOCKS ON THE FAILING PAIR OF OPERANDS
14270                                   ;;**************************************************************
14271  053404                           TST727:
14272  053404  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14273  053406  012700  000727                  MOV     #727,R0         ;;LOAD R0 WITH TEST NUMBER
14274  053412  005001                  10$:    CLR     R1              ;INITIALIZE REGS TO 000000
14275  053414  005002                          CLR     R2
14276  053416  005004                          CLR     R4
14277  053420  010737  001010                  MOV     PC,@#$LPERR     ;LOOP ONLY ON FAILING PAIR OF #'S
14278  053424  010203                  1$:     MOV     R2,R3           ;LOAD DEST OPERAND
14279  053426  000257                          CCC                     ;SCOPE SYNC
14280
14281  053430  060103                  2$:     ADD     R1,R3           ;ADD THE TWO TEST NO.S
14282                                                                  ;RESULT S / B = 000000
14283
14284  053432  020403                          CMP     R4,R3           ;RESULT = 000000 ?
14285  053434  001402                          BEQ     4$              ;BR IF YES
14286
14287  053436  104010                  3$:     ERROR   10              ;INCORRECT RESULT IN R3
14288
14289  053440  000407                          BR      TST730          ;;GO TO SCOPE EXIT
14290
14291  053442  005202                  4$:     INC     R2              ;ADD 1 TO DEST. OP
14292  053444  100402                          BMI     5$              ;GET OUT IF IT WENT NEGATIVE
14293  053446  005301                          DEC     R1              ;SUB 1 FROM THE SOURCE OP
14294  053450  000765                          BR      1$              ;GO ADD THE TWO NO.S
14295
14296  053452  012737  053412  001010  5$:     MOV     #10$,@#$LPERR   ;LOOP FROM BEGINNING ON ERROR
14297
14298                                   ;;**************************************************************
14299                                   ;*TEST 730      MUL RA,RB TEST ; N:C = 1111
14300                                   ;;**************************************************************
14301  053460                           TST730:
14302  053460  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14303  053462  012700  000730                  MOV     #730,R0         ;;LOAD R0 WITH TEST NUMBER
14304  053466  013737  053516  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14305  053474  005001                          CLR     R1              ;S/B RESULT IN R2
14306  053476  012704  000006                  MOV     #6,R4           ;S/B RESULT IN R3
14307  053502  012702  000002                  MOV     #2,R2           ;INITIALIZE REG
14308  053506  005003                          CLR     R3              ;INITIALIZE REG + 1
14309  053510  012705  000003                  MOV     #3,R5           ;INITIALIZE SRC
14310  053514  000277                          SCC                     ;SCOPE SYNC
14311
14312  053516  070205                  2$:     MUL     R5,R2           ;TEST THE MUL
14313
14314  053520  100403                          BMI     3$              ;N:C=0000?
14315  053522  001402                          BEQ     3$
14316  053524  102401                          BVS     3$
14317  053526  103001                          BCC     4$
14318
14319  053530  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14320
14321  053532  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14322  053534  001002                          BNE     5$              ;BR IF NOT
14323  053536  020102                          CMP     R1,R2           ;REG CORRECT?
```

```
14324  053540  001401                        BEQ     TST731          ;;BR IF YES
14325
14326  053542  104045                5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14327
14328                                  ;;*********************************************************************
14329                                  ;*TEST 731      MUL (RA),RB TEST ; N:C = 0000-SET C
14330                                  ;;*********************************************************************
14331  053544                          TST731:
14332  053544  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14333  053546  012700  000731                MOV     #731,R0         ;;LOAD R0 WITH TEST NUMBER
14334  053552  013737  053606  001076        MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14335  053560  005001                        CLR     R1              ;S/B RESULT IN R2
14336  053562  012704  123450                MOV     #123450,R4      ;S/B RESULT IN R3
14337  053566  012702  012345                MOV     #012345,R2      ;INITIALIZE REG
14338  053572  005003                        CLR     R3              ;INITIALIZE REG + 1
14339  053574  012705  063312                MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14340  053600  012715  000010                MOV     #10,(R5)        ;INITIALIZE SRC
14341  053604  000257                        CCC                     ;SCOPE SYNC
14342
14343  053606  070215                2$:     MUL     (R5),R2         ;TEST THE MUL
14344
14345  053610  100403                        BMI     3$              ;N:C-0001?
14346  053612  001402                        BEQ     3$
14347  053614  102401                        BVS     3$
14348  053616  103401                        BCS     4$
14349
14350  053620  104044                3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14351
14352  053622  020304                4$:     CMP     R3,R4           ;REG+1 CORRECT?
14353  053624  001002                        BNE     5$              ;BR IF NOT
14354  053626  020102                        CMP     R1,R2           ;REG CORRECT?
14355  053630  001401                        BEQ     TST732          ;;BR IF YES
14356
14357  053632  104045                5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14358
14359                                  ;;*********************************************************************
14360                                  ;*TEST 732      MUL (RA)+,RB TEST ; N:C   0000-SET Z
14361                                  ;;*********************************************************************
14362  053634                          TST732:
14363  053634  000004                        SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14364  053636  012700  000732                MOV     #732,R0         ;;LOAD R0 WITH TEST NUMBER
14365  053642  013737  053674  001076        MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14366  053650  005001                        CLR     R1              ;S/B RESULT IN R2
14367  053652  005004                        CLR     R4              ;S/B RESULT IN R3
14368  053654  005002                        CLR     R2              ;INITIALIZE REG
14369  053656  012703  177777                MOV     #-1,R3          ;INITIALIZE REG + 1
14370  053662  012705  063312                MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14371  053666  012715  000010                MOV     #10,(R5)        ;INITIALIZE SRC
14372  053672  000257                        CCC                     ;SCOPE SYNC
14373
14374  053674  070225                2$:     MUL     (R5)+,R2        ;TEST THE MUL
14375
14376  053676  100403                        BMI     3$              ;N:C=0100?
14377  053700  001002                        BNE     3$
14378  053702  102401                        BVS     3$
14379  053704  103001                        BCC     4$
```

```
14380
14381  053706  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14382
14383  053710  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14384  053712  001002                          BNE     5$              ;BR IF NOT
14385  053714  020102                          CMP     R1,R2           ;REG CORRECT?
14386  053716  001401                          BEQ     6$              ;BR IF YES
14387
14388  053720  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14389
14390  053722  022705  063314          6$:     CMP     #MBUF0+2,R5     ;DID R5 GET AUTO-INCREMENTED?
14391  053726  001401                          BEQ     TST733          ;;BR IF YES
14392
14393  053730  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14394
14395                                  ;;*******************************************************************
14396                                  ;*TEST 733       MUL @(RA)+,RB TEST ; N:C = 0000-SET N ; SRC,DST - -,+
14397                                  ;;*******************************************************************
14398  053732                          TST733:
14399  053732  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14400  053734  012700  000733                  MOV     #733,R0         ;;LOAD R0 WITH TEST NUMBER
14401  053740  013737  054000  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14402  053746  012701  177777                  MOV     #-1,R1          ;S/B RESULT IN R2
14403  053752  012704  177770                  MOV     #-10,R4         ;S/B RESULT IN R3
14404  053756  012702  000001                  MOV     #1,R2           ;INITIALIZE REG
14405  053762  005003                          CLR     R3              ;INITIALIZE REG + 1
14406  053764  012705  063306                  MOV     #ATA+10,R5      ;SET UP POINTER TO POINTER TO MBUF0
14407  053770  012737  177770  063312          MOV     #-10,@#MBUF0    ;INITIALIZE SRC
14408  053776  000257                          CCC                     ;SCOPE SYNC
14409
14410  054000  070235                  2$:     MUL     @(R5)+,R2       ;TEST THE MUL
14411
14412  054002  100003                          BPL     3$              ;N:C=1000?
14413  054004  001402                          BEQ     3$
14414  054006  102401                          BVS     3$
14415  054010  103001                          BCC     4$
14416
14417  054012  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14418
14419  054014  020304                  4$:     CMP     R3,R4           ;REG+1 CORRECT?
14420  054016  001002                          BNE     5$              ;BR IF NOT
14421  054020  020102                          CMP     R1,R2           ;REG CORRECT?
14422  054022  001401                          BEQ     6$              ;BR IF YES
14423
14424  054024  104045                  5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14425
14426  054026  022705  063310          6$:     CMP     #ATA+12,R5      ;DID R5 GET AUTO-INCREMENTED?
14427  054032  001401                          BEQ     TST734          ;;BR IF YES
14428
14429  054034  104046                          ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14430
14431                                  ;;*******************************************************************
14432                                  ;*TEST 734       MUL -(RA),RB TEST ; N:C = 1111-CLR ALL BUT N ; SRC,DSK - +,-
14433                                  ;;*******************************************************************
14434  054036                          TST734:
14435  054036  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
14436  054040  012700  000734                 MOV     #734,R0          ;;LOAD R0 WITH TEST NUMBER
14437  054044  013737  054104  001076         MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
14438  054052  012701  177777                 MOV     #-1,R1           ;S/B RESULT IN R2
14439  054056  012704  177770                 MOV     #-10,R4          ;S/B RESULT IN R3
14440  054062  012702  177777                 MOV     #-1,R2           ;INITIALIZE REG
14441  054066  005003                          CLR     R3               ;INITIALIZE REG + 1
14442  054070  012705  063314                 MOV     #MBUF0+2,R5      ;SET UP POINTER TO SRC
14443  054074  012737  000010  063312         MOV     #10,@#MBUF0      ;INITIALIZE SRC
14444  054102  000277                          SCC                      ;SCOPE SYNC
14445
14446  054104  070245                  2$:     MUL     -(R5),R2         ;TEST THE MUL
14447
14448  054106  100003                          BPL     3$               ;N:C=1000?
14449  054110  001402                          BEQ     3$
14450  054112  102401                          BVS     3$
14451  054114  103001                          BCC     4$
14452
14453  054116  104044                  3$:     ERROR   44               ;COND CODES SET IMPROPERLY
14454
14455  054120  020304                  4$:     CMP     R3,R4            ;REG+1 CORRECT?
14456  054122  001002                          BNE     5$               ;BR IF NOT
14457  054124  020102                          CMP     R1,R2            ;REG CORRECT?
14458  054126  001401                          BEQ     6$               ;BR IF YES
14459
14460  054130  104045                  5$:     ERROR   45               ;MUL DELIVERED WRONG RESULT
14461
14462  054132  022705  063312         6$:     CMP     #MBUF0,R5        ;DID SRC REG GET AUTO-DECREMENTED?
14463  054136  001401                          BEQ     TST735           ;;BR IF YES
14464
14465  054140  104046                          ERROR   46               ;AUTO DECREMENT DID NOT OCCUR
14466
14467                                  ;;*****************************************************************
14468                                  ;*TEST 735      MUL @-(RA),RB TEST ; N:C = 1111-CLR ALL BUT C ; SRC,DST = -,-
14469                                  ;;*****************************************************************
14470  054142                          TST735:
14471  054142  000004                          SCOPE                    ;CALL THE SCOPE LOOP UTILITY
14472  054144  012700  000735                 MOV     #735,R0          ;;LOAD R0 WITH TEST NUMBER
14473  054150  013737  054210  001076         MOV     @#2$,@#$TMP0     ;GET TEST INSTRUCTION WORD
14474  054156  005001                          CLR     R1               ;S/B RESULT IN R2
14475  054160  012704  106420                 MOV     #106420,R4       ;/S/B RESULT IN R3
14476  054164  012702  177776                 MOV     #-2,R2           ;INITIALIZE REG
14477  054170  012703  177777                 MOV     #-1,R3           ;INITIALIZE REG + 1
14478  054174  012705  063310                 MOV     #ATA+12,R5       ;SET UP POINTER TO POINTER TO MBUF0
14479  054200  012737  134570  063312         MOV     #-43210,@#MBUF)  ;INITIALIZE SRC
14480  054206  000277                          SCC                      ;SCOPE SYNC
14481
14482  054210  070255                  2$:     MUL     @-(R5),R2        ;TEST THE MUL
14483
14484  054212  100403                          BMI     3$               ;N:C=0001?
14485  054214  001402                          BEQ     3$
14486  054216  102401                          BVS     3$
14487  054220  103401                          BCS     4$
14488
14489  054222  104044                  3$:     ERROR   44               ;COND CODES SET IMPROPERLY
14490
14491  054224  020304                  4$:     CMP     R3,R4            ;REG+1 CORRECT?
```

```
14492  054226  001002                  BNE     5$              ;BR IF NOT
14493  054230  020102                  CMP     R1,R2           ;REG CORRECT?
14494  054232  001401                  BEQ     6$              ;BR IF YES
14495
14496  054234  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14497
14498  054236  022705  063306  6$:     CMP     #ATA+10,R5      ;DID R5 GET AUTO-DECREMENTED?
14499  054242  001401                  BEQ     TST736          ;;BR IF YES
14500
14501  054244  104046                  ERROR   46              ;AUTO INCREMENT DID NOT OCCUR
14502
14503                          ;;*****************************************************************
14504                          ;*TEST 736       MUL X(RA),RB TEST ; N:C = 1111 TO 0100
14505                          ;;*****************************************************************
14506  054246                  TST736:
14507  054246  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14508  054250  012700  000736          MOV     #736,R0         ;;LOAD R0 WITH TEST NUMBER
14509  054254  013737  054310  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14510  054262  005001                  CLR     R1              ;S/B RESULT IN R2
14511  054264  005004                  CLR     R4              ;S/B RESULT IN R3
14512  054266  012702  012345          MOV     #012345,R2      ;INITIALIZE REG
14513  054272  012703  177777          MOV     #-1,R3          ;INITIALIZE REG + 1
14514  054276  012705  063312          MOV     #MBUF0,R5       ;SET UP POINTER TO SRC
14515  054302  005065  000002          CLR     2(R5)           ;INITIALIZE SRC
14516  054306  000277                  SCC                     ;SCOPE SYNC
14517
14518  054310  070265  000002  2$:     MUL     2(R5),R2        ;TEST THE MUL
14519
14520  054314  100403                  BMI     3$              ;N:C=0100?
14521  054316  001002                  BNE     3$
14522  054320  102401                  BVS     3$
14523  054322  103001                  BCC     4$
14524
14525  054324  104044          3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14526
14527  054326  020304          4$:     CMP     R3,R4           ;REG+1 CORRECT?
14528  054330  001002                  BNE     5$              ;BR IF NOT
14529  054332  020102                  CMP     R1,R2           ;REG CORRECT?
14530  054334  001401                  BEQ     TST737          ;;BR IF YES
14531
14532  054336  104045          5$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14533
14534                          ;;*****************************************************************
14535                          ;*TEST 737       MUL @X(RA),RB TEST
14536                          ;;*****************************************************************
14537  054340                  TST737:
14538  054340  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14539  054342  012700  000737          MOV     #737,R0         ;;LOAD R0 WITH TEST NUMBER
14540  054346  013737  054404  001076  MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14541  054354  005001                  CLR     R1              ;S/B RESULT IN R2
14542  054356  012704  000100          MOV     #100,R4         ;S/B RESULT IN R3
14543  054362  012702  000010          MOV     #10,R2          ;INITIALIZE REG
14544  054366  005003                  CLR     R3              ;INITIALIZE REG + 1
14545  054370  012705  063276          MOV     #ATA,R5         ;GET POINTER TO TABLE OF POINTERS
14546  054374  012737  000010  063312  MOV     #10,@#MBUF0     ;INITIALIZE SRC
14547  054402  000257                  CCC                     ;SCOPE SYNC
```

```
14548
14549  054404  070275  000010          2$:     MUL     @10(R5),R2      ;TEST THE MUL
14550
14551  054410  020304                          CMP     R3,R4           ;REG+1 CORRECT?
14552  054412  001002                          BNE     3$              ;BR IF NOT
14553  054414  020102                          CMP     R1,R2           ;REG CORRECT?
14554  054416  001401                          BEQ     TST740          ;;BR IF YES
14555
14556  054420  104045                  3$:     ERROR   45              ;MUL DELIVERED WRONG RESULT
14557
14558                                  ;;*****************************************************************
14559                                  ;*TEST 740      DIV #N,RA TEST ; N:C = 1111
14560                                  ;;*****************************************************************
14561  054422                          TST740:
14562  054422  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14563  054424  012700  000740                  MOV     #740,R0         ;;LOAD R0 WITH TEST NUMBER
14564  054430  013737  054456  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14565  054436  012701  010000                  MOV     #010000,R1      ;S/B RES IN R2
14566  054442  012704  000001                  MOV     #1,R4           ;S/B RES IN R3
14567  054446  005002                          CLR     R2              ;SET UP REG OPERAND
14568  054450  012703  020001                  MOV     #020001,R3      ;SET UP REG+1 OP
14569  054454  000277                          SCC                     ;SCOPE SYNC
14570
14571  054456  071227  000002          2$:     DIV     #2,R2           ;TEST DIV
14572
14573  054462  100403                          BMI     3$              ;N:C=0000?
14574  054464  001402                          BEQ     3$
14575  054466  102401                          BVS     3$
14576  054470  103001                          BCC     4$
14577
14578  054472  104044                  3$:     ERROR   44              ;COND CODES SET IMPROPERLY
14579
14580  054474  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14581  054476  001002                          BNE     5$              ;BR IF NOT
14582  054500  020102                          CMP     R1,R2           ;CORRECT RESULT IN REG?
14583  054502  001401                          BEQ     TST741          ;;BR IF YES
14584
14585  054504  104045                  5$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14586
14587                                  ;;*****************************************************************
14588                                  ;*TEST 741      DIV #N,RA TEST ; RA NEGATIVE ; N:C = 0000
14589                                  ;;*****************************************************************
14590  054506                          TST741:
14591  054506  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14592  054510  012700  000741                  MOV     #741,R0         ;;LOAD R0 WITH TEST NUMBER
14593  054514  013737  054544  001076          MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14594  054522  012701  177775                  MOV     #-3,R1          ;S/B RES IN R2
14595  054526  012704  177776                  MOV     #-2,R4          ;S/B RES IN R3
14596  054532  012702  177777                  MOV     #-1,R2          ;SET UP REG OPERAND
14597  054536  012703  177762                  MOV     #-14.,R3        ;SET UP REG+1 OP
14598  054542  000257                          CCC                     ;SCOPE SYNC
14599
14600  054544  071227  000004          2$:     DIV     #4,R2           ;TEST DIV
14601
14602  054550  100003                          BPL     3$              ;N:C=1000?
14603  054552  001402                          BEQ     3$
```

```
14604  054554  102401                    BVS     3$
14605  054556  103001                    BCC     4$
14606
14607  054560  104044              3$:   ERROR   44              ;COND CODES SET IMPROPERLY
14608
14609  054562  020304              4$:   CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14610  054564  001002                    BNE     5$              ;BR IF NOT
14611  054566  020102                    CMP     R1,R2           ;CORRECT RESULT IN REG?
14612  054570  001401                    BEQ     TST742          ;;BR IF YES
14613
14614  054572  104045              5$:   ERROR   45              ;DIV DELIVERED WRONG RESULT
14615
14616                                    ;;*************************************************************
14617                                    ;*TEST 742      DIV #N,RA TEST ; N:C = 0000 TO 0100
14618                                    ;;*************************************************************
14619  054574                            TST742:
14620  054574  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14621  054576  012700  000742            MOV     #742,R0         ;;LOAD R0 WITH TEST NUMBER
14622  054602  013737  054626  001076    MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14623  054610  005001                    CLR     R1              ;S/B RES IN R2
14624  054612  012704  000001            MOV     #1,R4           ;S/B RES IN R3
14625  054616  005002                    CLR     R2              ;SET UP REG OPERAND
14626  054620  012703  000001            MOV     #1,R3           ;SET UP REG+1 OP
14627  054624  000257                    CCC                     ;SCOPE SYNC
14628
14629  054626  071227  000002      2$:   DIV     #2,R2           ;TEST DIV
14630
14631  054632  100403                    BMI     3$              ;N:C=0100?
14632  054634  001002                    BNE     3$
14633  054636  102401                    BVS     3$
14634  054640  103001                    BCC     4$
14635
14636  054642  104044              3$:   ERROR   44              ;COND CODES SET IMPROPERLY
14637
14638  054644  020304              4$:   CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14639  054646  001002                    BNE     5$              ;BR IF NOT
14640  054650  020102                    CMP     R1,R2           ;CORRECT RESULT IN REG?
14641  054652  001401                    BEQ     TST743          ;;BR IF YES
14642
14643  054654  104045              5$:   ERROR   45              ;DIV DELIVERED WRONG RESULT
14644
14645                                    ;;*************************************************************
14646                                    ;*TEST 743      DIV #-N,RA TEST ; RA POS
14647                                    ;;*************************************************************
14648  054656                            TST743:
14649  054656  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14650  054660  012700  000743            MOV     #743,R0         ;;LOAD R0 WITH TEST NUMBER
14651  054664  013737  054712  001076    MOV     @#2$,@#$TMP0    ;GET COPY OF TEST INSTRUCTION
14652  054672  012701  177775            MOV     #-3,R1          ;S/B RES IN R2
14653  054676  012704  000002            MOV     #2,R4           ;S/B RES IN R3
14654  054702  005002                    CLR     R2              ;SET UP REG OPERAND
14655  054704  012703  000016            MOV     #14.,R3         ;SET UP REG+1 OP
14656  054710  000257                    CCC                     ;SCOPE SYNC
14657
14658  054712  071227  177774      2$:   DIV     #-4,R2          ;TEST DIV
14659
```

```
14660  054716  020304                  CMP     R3,R4           ;CORRECT RESULT IN REG+1?
14661  054720  001002                  BNE     3$              ;BR IF NOT
14662  054722  020102                  CMP     R1,R2           ;CORRECT RESULT IN REG?
14663  054724  001401                  BEQ     TST744          ;;BR IF YES
14664
14665  054726  104045          3$:     ERROR   45              ;DIV DELIVERED WRONG RESULT
14666
14667                          ;;*****************************************************************
14668                          ;*TEST 744      DIV  TEST - V BIT GETS SET
14669                          ;*      THIS TEST TESTS THAT THE V BIT CAN BE SET IN ALL THE
14670                          ;*      POSSIBLE WAYS.  SINCE THE INSTRUCTION SHOULD BE ABORTED, THE
14671                          ;*      RESULTS CANNOT BE GUARANTEED.  FOR THIS REASON, ONLY
14672                          ;*      THE CONDITION CODES ARE CHECKED.
14673                          ;;*****************************************************************
14674  054730                  TST744:
14675  054730  000004                  SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14676  054732  012700  000744          MOV     #744,R0         ;;LOAD R0 WITH TEST NUMBER
14677  054736  013701  054756          MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14678  054742  012704  000002          MOV     #2,R4           ;S/B PSW
14679  054746  005037  177776          CLR     @#PSW           ;CLEAR OUT OTHER PSW BITS
14680  054752  012702  000050          MOV     #50,R2          ;SET UP REG OP
14681
14682  054756  071227  000005  2$:     DIV     #5,R2           ;TEST DIV -- SHOULD ABORT
14683
14684  054762  100424                  BMI     3$              ;N:C=0010?
14685  054764  001423                  BEQ     3$
14686  054766  102022                  BVC     3$
14687  054770  103421                  BCS     3$
14688
14689  054772  012702  177777          MOV     #-1,R2          ;INITIALIZE REG OP
14690  054776  005003                  CLR     R3              ;INITILAIZE REG+1 OP
14691
14692  055000  071227  177776          DIV     #-2,R2          ;TEST DIV -- SHOULD ABORT
14693
14694  055004  100413                  BMI     3$              ;N:C=0010?
14695  055006  001412                  BEQ     3$
14696  055010  102011                  BVC     3$
14697  055012  103410                  BCS     3$
14698
14699  055014  012704  000003          MOV     #3,R4           ;S/B PSW
14700
14701  055020  071227  000000          DIV     #0,R2           ;TEST DIV BY 0 -- SHOULD ABORT
14702
14703  055024  100403                  BMI     3$              ;N:C=0010?
14704  055026  001402                  BEQ     3$
14705  055030  102001                  BVC     3$
14706  055032  103405                  BCS     TST745          ;;IF ALL OK, THEN EXIT TEST
14707
14708  055034  013703  177776  3$:     MOV     @#PSW,R3        ;GET WAS PSW
14709  055040  012702  177776          MOV     #PSW,R2         ;DESTINATION IS PSW
14710
14711  055044  104001                  ERROR   1               ;CONDITION CODES SET WRONG
14712
14713                          ;;*****************************************************************
14714                          ;*TEST 745      ASH #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14715                          ;;*****************************************************************
```

```
14716  055046                          TST745:
14717  055046  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14718  055050  012700  000745                  MOV     #745,R0         ;;LOAD R0 WITH TEST NUMBER
14719  055054  013701  055072                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14720  055060  012704  123450                  MOV     #123450,R4      ;S/B RESULT
14721  055064  012703  112345                  MOV     #112345,R3      ;INITIAL REG
14722  055070  000257                          CCC                     ;SCOPE SYNC
14723
14724  055072  072327  000003          2$:     ASH     #3,R3           ;TEST THE ASH
14725
14726  055076  100003                          BPL     3$              ;N:C=1010?
14727  055100  001402                          BEQ     3$
14728  055102  102001                          BVC     3$
14729  055104  103001                          BCC     4$
14730
14731  055106  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14732
14733  055110  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14734  055112  001401                          BEQ     TST746          ;;BR IF YES
14735  055114  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14736
14737                                  ;;*********************************************************************
14738                                  ;*TEST 746       ASH #N,RA TEST ; SHIFT RIGHT ; N:C = 1000 TO 0101
14739                                  ;;*********************************************************************
14740  055116                          TST746:
14741  055116  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14742  055120  012700  000746                  MOV     #746,R0         ;;LOAD R0 WITH TEST NUMBER
14743  055124  013701  055142                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14744  055130  005004                          CLR     R4              ;S/B RESULT
14745  055132  012703  000004                  MOV     #4,R3           ;INITIAL REG
14746  055136  000257                          CCC                     ;SCOPE SYNC
14747  055140  000270                          SEN                     ;CODES = 1000
14748
14749  055142  072327  177775          2$:     ASH     #-3,R3          ;TEST THE ASH
14750
14751  055146  100403                          BMI     3$              ;N:C=0101?
14752  055150  001002                          BNE     3$
14753  055152  102401                          BVS     3$
14754  055154  103401                          BCS     4$
14755
14756  055156  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14757
14758  055160  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14759  055162  001401                          BEQ     TST747          ;;BR IF YES
14760  055164  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14761
14762                                  ;;*********************************************************************
14763                                  ;*TEST 747       ASH #N,RA TEST ; SHIFT LEFT ; N:C = 1111 TO 1000
14764                                  ;;*********************************************************************
14765  055166                          TST747:
14766  055166  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14767  055170  012700  000747                  MOV     #747,R0         ;;LOAD R0 WITH TEST NUMBER
14768  055174  013701  055212                  MOV     @#2$,R1         ;LOAD R1 WITH TEST INSTRUCTION WORD
14769  055200  012704  177234                  MOV     #177234,R4      ;S/B RESULT
14770  055204  012703  123432                  MOV     #123432,R3      ;INITIAL REG
14771  055210  000277                          SCC                     ;SCOPE SYNC
```

```
14772
14773  055212  072327  177772          2$:     ASH     #-6,R3          ;TEST THE ASH
14774
14775  055216  100003                          BPL     3$              ;N:C=1000?
14776  055220  001402                          BEQ     3$
14777  055222  102401                          BVS     3$
14778  055224  103001                          BCC     4$
14779
14780  055226  104002                  3$:     ERROR   2               ;INCORRECT CONDITION CODES
14781
14782  055230  020304                  4$:     CMP     R3,R4           ;CORRECT RESULT?
14783  055232  001401                          BEQ     TST750          ;;BR IF YES
14784  055234  104002                          ERROR   2               ;ASH DELIVERED WRONG RESULT
14785
14786                                  ;;*********************************************************************
14787                                  ;*TEST 750       ASHC #N,RA TEST ; SHIFT LEFT ; N:C = 0000 TO 1010
14788                                  ;;*********************************************************************
14789  055236                          TST750:
14790  055236  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14791  055240  012700  000750                  MOV     #750,R0         ;;LOAD R0 WITH TEST NUMBER
14792  055244  013737  055274  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14793  055252  012701  123456                  MOV     #123456,R1      ;S/B RES IN R2
14794  055256  012704  076530                  MOV     #076530,R4      ;S/B RES IN R3
14795  055262  012702  112345                  MOV     #112345,R2      ;INITIALIZE COMBINED
14796  055266  012703  147653                  MOV     #147653,R3      ;  REGISTERS
14797  055272  000257                          CCC                     ;SCOPE SYNC
14798
14799  055274  073227  000003          2$:     ASHC    #3,R2           ;TEST ASHC
14800
14801  055300  100003                          BPL     3$              ;N:C=1010?
14802  055302  001402                          BEQ     3$
14803  055304  102001                          BVC     3$
14804  055306  103001                          BCC     4$
14805
14806  055310  104044                  3$:     ERROR   44              ;COND CODES WRONG
14807
14808  055312  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14809  055314  001002                          BNE     5$              ;BR IF NOT
14810  055316  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14811  055320  001401                          BEQ     TST751          ;;BR IF YES
14812  055322  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14813
14814                                  ;;*********************************************************************
14815                                  ;*TEST 751       ASHC #N,RA TEST ; SHIFT RIGHT ; N:C - 1000 TO 0101
14816                                  ;;*********************************************************************
14817  055324                          TST751:
14818  055324  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14819  055326  012700  000751                  MOV     #751,R0         ;;LOAD R0 WITH TEST NUMBER
14820  055332  013737  055356  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14821  055340  005001                          CLR     R1              ;S/B RES IN R2
14822  055342  005004                          CLR     R4              ;S/B RES IN R3
14823  055344  005002                          CLR     R2              ;INITIALIZE COMBINED
14824  055346  012703  000005                  MOV     #5,R3           ;  REGISTERS
14825  055352  000257                          CCC                     ;SCOPE SYNC
14826  055354  000270                          SEN                     ;CODES = 1000
14827
```

```
14828  055356  073227  177775          2$:     ASHC    #-3,R2          ;TEST ASHC
14829
14830  055362  100403                          BMI     3$              ;N:C=0101?
14831  055364  001002                          BNE     3$
14832  055366  102401                          BVS     3$
14833  055370  103401                          BCS     4$
14834
14835  055372  104044                  3$:     ERROR   44              ;COND CODES WRONG
14836
14837  055374  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14838  055376  001002                          BNE     5$              ;BR IF NOT
14839  055400  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14840  055402  001401                          BEQ     TST752          ;;BR IF YES
14841  055404  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14842
14843                                  ;;*****************************************************************
14844                                  ;*TEST 752       ASHC #N,RA TEST : SHIFT RIGHT ; N:C = 1111 TO 1000
14845                                  ;;*****************************************************************
14846  055406                          TST752:
14847  055406  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14848  055410  012700  000752                  MOV     #752,R0         ;;LOAD R0 WITH TEST NUMBER
14849                                  .SBTTL USER CONTROLLED BREAKPOINT -- BIT15
14850  055414  032737  100000  063234          BIT     #BIT15,@#BPTLOC ;BREAKPOINT HALT SET ??
14851  055422  001401                          BEQ     .+4             ;BR IF NOT
14852  055424  000000                          HALT                    ;BREAK-DEPRESS CONTINUE TO CONTINUE
14853  055426  013737  055456  001076          MOV     @#2$,@#$TMP0    ;GET TEST INSTRUCTION WORD
14854  055434  012701  177234                  MOV     #177234,R1      ;S/B RES IN R2
14855  055440  012704  135275                  MOV     #135275,R4      ;S/B RES IN R3
14856  055444  012702  123456                  MOV     #123456,R2      ;INITIALIZE COMBINED
14857  055450  012703  127542                  MOV     #127542,R3      ;  REGISTERS
14858  055454  000257                          CCC                     ;SCOPE SYNC
14859
14860  055456  073227  177772          2$:     ASHC    #-6,R2          ;TEST ASHC
14861
14862  055462  100003                          BPL     3$              ;N:C=1000?
14863  055464  001402                          BEQ     3$
14864  055466  102401                          BVS     3$
14865  055470  103401                          BCS     4$
14866
14867  055472  104044                  3$:     ERROR   44              ;COND CODES WRONG
14868
14869  055474  020102                  4$:     CMP     R1,R2           ;TOP HALF OF RESULT CORRECT?
14870  055476  001002                          BNE     5$              ;BR IF NOT
14871  055500  020403                          CMP     R4,R3           ;LOWER HALF OF RESULT CORRECT?
14872  055502  001401                          BEQ     TST753          ;;BR IF YES
14873  055504  104045                  5$:     ERROR   45              ;ASHC DELIVERED WRONG RES
14874
14875                                  ;*
14876                                  ;*      THIS SECTION OF THE MED TESTS EXERCISES CERTAIN SCRATCH
14877                                  ;*      PAD REGISTERS USING MED READS AND WRITES.  THEIR ORIGINAL
14878                                  ;*      CONTENTS ARE RESTORED BUT:
14879                                  ;*
14880                                  ;*      ******************** IMPORTANT NOTE ********************
14881                                  ;*      *                                                       *
14882                                  ;*      *  THE CONSOLE MUST NOT !!! BE USED DURING THESE MED    *
14883                                  ;*      *  TESTS. NO INTERRUPTS OR TRAPS CAN BE ALLOWED EITHER  *
```

```
14884                                   ;*      *                                                       *
14885                                   ;*      *****************************************************
14886                                   ;*
14887
14888                                   ;;*****************************************************************
14889                                   ;*TEST 753      CHECK MED IS ILLEGAL IN USER - EXECUTES IN KERNAL
14890                                   ;*      THE NEXT TEST BELOW CHECKS TO SEE THAT THE "MED"
14891                                   ;*      (MAINTENANCE, EXAM, AND DEPOSIT) INSTRUCTION WILL EXECUTE
14892                                   ;*      WHEN IN KERNEL MODE WITHOUT AFFECTING THE PSW AND
14893                                   ;*      THAT IT IS ILLEGAL IN USER MODE
14894                                   ;;*****************************************************************
14895  055506                           TST753:
14896  055506  012700  000752                  MOV     #752,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14897  055512  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14898  055514  012737  000304  177770  MED1:   MOV     #304,@#UBREAK   ;SET SCOPE SYNC FOR MED INSTR
14899  055522  012737  140000  177776          MOV     #140000,@#PSW   ;GO TO USER MODE
14900  055530  012706  001000                  MOV     #STACK,SP       ;SETUP USER STACK PTR.
14901  055534  012737  055566  000004          MOV     #2$,@#ERRVEC    ;SET ERROR TRAP VECTOR TO 2$ BELOW
14902  055542  012737  055566  000010          MOV     #2$,@#RESVEC    ;LOAD RESERVED INST. TRAP VECTOR
14903  055550  012701  177777                  MOV     #-1,R1          ;LOAD R1 WITH A -1
14904  055554  005000                          CLR     R0              ;CLEAR R0
14905  055556  076600                          MED                     ;TRY TO DO MAINT. EXAMINE
14906  055560  000041                          .WORD   041             ;MED READ CODE FOR R1
14907  055562  104012                          ERROR   12              ;ERROR - MED INST. NOT ILLEGAL IN USER
14908  055564  000404                          BR      4$
14909  055566  005700                  2$:     TST     R0              ;IS R0 UNCHANGED?
14910  055570  001401                          BEQ     3$              ;BRANCH IF YES
14911  055572  104013                          ERROR   13              ;ERROR - MED INSTRUCTION WAS EXECUTED
14912                                                                  ;BEFORE TRAPPING
14913  055574  022626                  3$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
14914  055576  012737  061220  000004  4$:     MOV     #BERR,@#ERRVEC  ;RESTORE ERROR TRAP VECTOR
14915  055604  012737  061122  000010          MOV     #RSERR,@#RESVEC ;RESTORE RESERVED INST. TRAP VECTOR
14916
14917  055612  005037  177776          MED0:   CLR     @#PSW           ;GO TO KERNEL MODE,CLEAR COND. CODES
14918  055616  076600                          MED                     ;DO MAINT. EXAMINE OF R1
14919  055620  000041                          .WORD   041             ;MED READ CODE FOR R1
14920  055622  103403                          BCS     MEDHLT
14921  055624  102402                          BVS     MEDHLT
14922  055626  100401                          BMI     MEDHLT
14923  055630  001001                          BNE     .+4
14924  055632  104014                  MEDHLT: ERROR   14              ;ERROR CC-BITS IN PSW AFFECTED BY MED
14925
14926                                   ;;*****************************************************************
14927                                   ;*TEST 754      MED TEST - R/W DATA PATTERNS TO REGS
14928                                   ;*      THIS PARTICULAR MED TEST WRITES DATA PATTERNS
14929                                   ;*      TO THOSE INTERNAL REGS. WHICH CAN BE WRITTEN
14930                                   ;*      AND READ WITHOUT SPECIAL CONSIDERAIONS.  REGISTERS
14931                                   ;*      REQUIRING SPECIAL TESTS ARE TESTED IN LATER
14932                                   ;*      MED TESTS.
14933                                   ;*      TABLE II CONTAINS THE REGISTER ADDRESSES.
14934                                   ;*
14935                                   ;*      A MAX. OF 3 ERRORS ARE REPORTED FOR EACH LOC.
14936                                   ;*
14937                                   ;;*****************************************************************
14938  055634                           TST754:
14939  055634  012700  000753                  MOV     #753,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
```

```
14940  055640  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14941  055642  012737  000340  177776  MEDT1:  MOV     #340,@#PSW      ;KERNEL MODE-PRIORITY 7
14942  055650  012701  064166                  MOV     #TBL2,R1        ;INITIALIZE ADDRESS POINTER
14943  055654  012737  125252  001102  1$:     MOV     #125252,@#$TMP2
14944  055662  111137  055730                  MOVB    (R1),@#11$      ;PUT WRITE CODE BY 'WRITE-MED'S"
14945  055666  112137  055752                  MOVB    (R1)+,@#13$     ;AND POINT R1 TO READ CODE
14946  055672  111137  055710                  MOVB    (R1),@#10$      ;PUT READ CODE BY "READ-MED'S"
14947  055676  112137  055736                  MOVB    (R1)+,@#12$     ;R1 NOW POINTS TO NEXT REG.
14948  055702  005037  001106                  CLR     @#$TMP4         ;CLEAR ERROR COUNTER
14949  055706  076600                  2$:     MED                     ;MED-READ THE INTERNAL REG.
14950  055710  000000                  10$:    .WORD   0               ;MED-READ CODE
14951  055712  010037  001076                  MOV     R0,@#$TMP0      ;SAVE ITS ORIGINAL CONTENTS
14952  055716  010137  001100                  MOV     R1,@#$TMP1      ;SAVE ADDR. PTR. VALUE
14953  055722  013700  001102                  MOV     @#$TMP2,R0      ;LOAD R0 WITH DATA TO BE WRITTEN
14954  055726  076600                          MED                     ;MED-WRITE THE TEST DATA
14955  055730  000000                  11$:    .WORD   0               ;MED-WRITE CODE
14956  055732  005000                          CLR     R0              ;CLEAR R0
14957  055734  076600                          MED                     ;MED-READ THE DATA BACK
14958  055736  000000                  12$:    .WORD   0               ;MED-READ CODE
14959  055740  010037  001104                  MOV     R0,@#$TMP3      ;SAVE DATA READ FOR COMPARISON
14960  055744  013700  001076                  MOV     @#$TMP0,R0      ;LOAD ORIGINAL DATA IN R0
14961  055750  076600                          MED                     ;MED-WRITE ORG. DATA TO REG.
14962  055752  000000                  13$:    .WORD   0               ;MED-WRITE CODE
14963  055754  023737  001102  001104          CMP     @#$TMP2,@#$TMP3 ;DID DATA READ=DATA WRITTEN?
14964  055762  001412                          BEQ     3$              ;BRANCH IF YES
14965  055764  013737  055736  001100          MOV     @#12$,@#$TMP1   ;SAVE MED-CODE FOR ERROR
14966  055772  022737  000003  001106          CMP     #3,@#$TMP4      ;MAX. ERROR REPORTS YET?
14967  056000  002401                          BLT     14$             ;BRANCH IF YES
14968  056002  104022                          ERROR   22              ;INT. REG. READ BACK WRONG DATA
14969  056004  005237  001106          14$:    INC     @#$TMP4         ;INCREMENT ERROR COUNTER
14970  056010  005137  001102          3$:     COM     @#$TMP2         ;CHANGE DATA PATTERN
14971  056014  013701  001100                  MOV     @#$TMP1,R1      ;RESTORE ADDR. POINTER
```

```
14972  056020  022737  125252  001102          CMP     #125252,@#$TMP2 ;BOTH DATA PATTERNS BEEN USED?
14973  056026  001327                          BNE     2$              ;BRANCH IF NO
14974  056030  005711                          TST     (R1)            ;END OF ADDR. TABLE?
14975  056032  001310                          BNE     1$              ;BRANCH IF NO
14976
14977                                  ;;*****************************************************************
14978                                  ;*TEST 755       MED TEST - VERIFY NOPS; READ R7 IN A & B SP
14979                                  ;*
14980                                  ;*      THIS TEST CHECKS ALL OF THE "NOP" OPERATION CODES
14981                                  ;*      TO ENSURE THEY WILL EXECUTE AS NOP'S AND
14982                                  ;*      NOT RESULT IN A PROCESSOR HANG.  THE "NOPS"
14983                                  ;*      TABLE (TABLE III) HOLDS THESE CODES.
14984                                  ;*      THIS TEST ALSO READS THE PROGRAM COUNTER (R7) VALUES
14985                                  ;*      STORED IN A & B SCRATCH PADS TO SEE THAT THEY
14986                                  ;*      READ PROPERLY.  THE R7 ADDRESSES ARE IN TABLE IV.
14987                                  ;*
14988                                  ;;*****************************************************************
14989  056034                          TST755:
14990  056034  012700  000754                 MOV     #754,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
14991  056040  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
14992  056042  012701  064404          MEDT3: MOV     #TBL3,R1        ;INITIALIZE NOP TABLE PTR. (R1)
14993  056046  112137  056054          1$:    MOVB    (R1)+,@#10$     ;PLACE FIRST "NOP-CODE" AFTER MED
14994                                                                  ;AND POINT R1 TO LAST CODE IN GROUP
14995  056052  076600                  5$:    MED                     ;EXECUTE MED WITH NOP OP-CODE
14996  056054  000000                  10$:   .WORD   0
14997  056056  123711  056054                 CMPB    @#10$,(R1)      ;HAVE ALL NOPS IN THAT GROUP
14998                                                                  ;BEEN TESTED?
14999  056062  103003                          BHIS    6$              ;BRANCH IF YES
15000  056064  005237  056054                 INC     @#10$           ;NEXT NOP IN GROUP
15001  056070  000770                          BR      5$
15002  056072  105721                  6$:    TSTB    (R1)+           ;POINT R1 TO NEXT NOP GROUP
15003  056074  005711                          TST     (R1)            ;HAVE ALL GROUPS BEEN TESTED
15004  056076  001363                          BNE     1$              ;BRANCH IF NO
15005
15006  056100  113737  064425  056112  MEDT4: MOVB    @#R7A+1,@#5$    ;LOAD R7A READ CODE AFTER MED
15007  056106  005000                  4$:    CLR     R0              ;CLEAR R0
15008  056110  076600                          MED                     ;MED READ R7 IN THE ASP
15009  056112  000000                  5$:    .WORD   0               ;READ CODE FOR R7A
15010  056114  020027  056114                 CMP     R0,#5$+2        ;DID R7A READ CORRECTLY?
15011  056120  001411                          BEQ     6$              ;BRANCH IF YES
15012  056122  013737  056112  001100          MOV     @#5$,@#$TMP1    ;SAVE MED-CODE FOR ERROR
15013  056130  012737  056114  001102          MOV     #5$+2,@#$TMP2   ;SAVE DATA EXPECTED
15014  056136  010037  001104                 MOV     R0,@#$TMP3      ;SAVE DATA RECEIVED
15015  056142  104022                          ERROR   22              ;R7A DID NOT READ THE RIGHT VALUE
15016  056144  023727  056112  000047  6$:    CMP     @#5$,#47        ;HAS R7B BEEN CHECKED?
15017  056152  001404                          BEQ     8$              ;BRANCH IF YES
15018  056154  113737  064431  056112          MOVB    @#R7B+1,@#5$    ;LOAD R7B READ CODE AFTER MED
15019  056162  000751                          BR      4$              ;TEST R7 BSP
15020  056164                          8$:
15021
15022
15023                                  ;;*****************************************************************
15024                                  ;*TEST 756       MED TEST - CSP CONSTANTS CHECK
15025                                  ;*
15026                                  ;*      THIS TEST CHECKS THE CONSTANT VALUES LOCATED
15027                                  ;*      IN THE C SCRATCH PAD.  THE CONSTANTS ARE READ
```

```
15028                                   ;*      WITH A MED INSTRUCTION AND COMPARED TO THEIR
15029                                   ;*      EXPECTED VALUE.  THE ADDRESSES OF THESE CONSTANTS
15030                                   ;*      AND THE VALUES EXPECTED ARE IN TABLE VII.
15031                                   ;*
15032                                   ;;*****************************************************************
15033  056164                           TST756:
15034  056164  012700  000755                  MOV     #755,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15035  056170  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15036
15037  056172  076600                          MED
15038  056174  000144                          RDFLAG
15039  056176  052700  004000                  BIS     #BIT11,R0       ;SET THE "CSP INVALID BIT" IN FLAG REG.
15040  056202  076600                          MED
15041  056204  000344                          WRFLAG
15042  056206  170000                  MEDT10: CFCC                    ;EXECUTE FLT. PT INST. SO FLT. PT.
15043                                           ;CONSTANTS ARE LOADED INTO CSP
15044  056210  012701  064536                  MOV     #TBL7,R1        ;SETUP TABLE POINTER
15045  056214  012167  000006          10$:    MOV     (R1)+,1$        ;LOAD MED READ CODE AT 1$
15046  056220  001414                          BEQ     11$             ;BR IF END OF TABLE
15047  056222  005000                          CLR     R0
15048  056224  076600                          MED                     ;READ INTERNAL CONTENTS INTO R0
15049  056226  000000                  1$:     .WORD   0
15050  056230  020021                          CMP     R0,(R1)+        ;WAS THE CONSTANT READ THE ONE EXPECTED
15051  056232  001770                          BEQ     10$             ;BRANCH IF YES
15052  056234  013737  056226  001100          MOV     @#1$,@#$TMP1    ;SAVE MEDCODE FOR ERROR
15053  056242  016137  177776  001102          MOV     -2(R1),@#$TMP2  ;SAVE CONSTANT VALUE EXPECTED
15054  056250  104021                          ERROR   21              ;CSP LOCATION HELD WRONG VALUE
15055  056252                           11$:
15056
15057
15058                                   ;;*****************************************************************
15059                                   ;*TEST 757      MED TEST - MICROBK CHECK OF MICRO-POINTS
15060                                   ;*
15061                                   ;*      THIS TEST USES THE MICROBREAK REGISTER AND THE
15062                                   ;*      INFORMATION IN TABLE V TO CHECK THAT THE
15063                                   ;*      CORRECT MED-FLOW IS ENTERED WHEN EACH
15064                                   ;*      REGISTER IS ACCESSED BY A MED INSTRUCTION.
15065                                   ;*      THE MICROBREAK REG. IS SETUP TO CAUSE A TRAP TO
15066                                   ;*      LOC. 4 WHEN ITS CONTENTS EQUAL THE ADDRESS
15067                                   ;*      OF THE MIRCOWORD BEING EXECUTED.
15068                                   ;*
15069                                   ;*          NOTE:  THE MICRO BREAK - TRAP-TO-4 CAPABILITY
15070                                   ;*                 IS TRIED AT THE BEGINNING OF THE TEST.
15071                                   ;*                 IF IT DOESN'T WORK, AN ERROR IS PRINTED
15072                                   ;*                 AND THE TEST IS SKIPPED
15073                                   ;*
15074                                   ;;*****************************************************************
15075  056252                           TST757:
15076  056252  012700  000756                  MOV     #756,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15077  056256  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15078  056260  012737  000071  177770  MEDT11: MOV     #SWB01,@#UBREAK ;LOAD MICROBK. REG. WITH AN MICRO ADDR.
15079  056266  012737  061104  000004          MOV     #BKROUT,@#4     ;LOAD ADDR. OF MICROBK. ROUTINE IN 4
15080  056274  012737  000340  000006          MOV     #340,@#6        ;LOAD KERNEL PSW - PRIORITY 7 IN 6
15081  056302  005037  061112                  CLR     @#BKFLAG        ;CLEAR MICROBK. TRAP FLAG
15082  056306  076600                          MED                     ;GET WHAMI INTO R0
15083  056310  000022                          RDWHAMI
```

```
15084  056312  052700  001000          BIS     #BIT9,R0        ;SET BIT 9
15085  056316  076600                  MED                     ;MED-WRITE THE WHAMI REG TO
15086  056320  000222          10$:    WRWHAMI                 ;ENABLE MICROBK-TRAP-TO-4
15087  056322  076600                  MED                     ;GET FLAG REGISTER
15088  056324  000144                  RDFLAG
15089  056326  052700  100000          BIS     #BIT15,R0       ;SET BIT 15 IN R0
15090  056332  076600                  MED                     ;MED-WRITE THE FLAG REG TO
15091  056334  000344          11$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15092  056336  000300                  SWAB    R0              ;MICROBK TRAP SHOULD OCCUR ON SWAB
15093  056340  005737  061112          TST     @#BKFLAG        ;DID TRAP TO 4 OCCUR?
15094  056344  001007                  BNE     1$              ;BRANCH IF YES
15095  056346  005037  001076          CLR     @#$TMP0
15096  056352  016737  121513  001100  MOV     SWB01,@#$TMP1   ;SAVE EXPECTED UBREAK ADDR
15097  056360  104015                  ERROR   15              ;MICROBREAK TRAP DIDN'T WORK
15098  056362  000453                  BR      50$             ;SKIP TO END OF TEST
15099
15100  056364  012701  000710          1$:     MOV     #SWB01+10,R1    ;GET CORRECT U-ADDR
15101  056370  076600                  MED                     ;GET LOG CUA REG
15102  056372  000103                  RDLCUA
15103  056374  042700  100007          BIC     #100007,R0      ;GET RID OF IRRELEVANT BITS
15104  056400  020001                  CMP     R0,R1           ;WAS CORRECT UADDR LOGGED?
15105  056402  001401                  BEQ     3$              ;BR IF YES
15106  056404  104025                  ERROR   25              ;CUA CONTAINS INCORRECT U-ADDR
15107  056406  012701  064436          3$:     MOV     #TBL5,R1                ;INTIALIZE TABLE PTR. (R1)
15108  056412  012702  064464          MOV     #TBL6,R2
15109  056416  010737  001010          MOV     PC,@#$LPERR     ;SET ERROR LOOP RETURN TO 2$
15110  056422  111137  056460          2$:     MOVB    (R1),@#12$      ;LOAD WRITE CODE AFTER MED
15111  056426  001431                  BEQ     50$             ;BR IF END OF TABLE
15112  056430  011237  177770          4$:     MOV     (R2),@#UBREAK   ;LOAD MICROBK REG. WITH MICROADDR.
15113  056434  005037  061112          CLR     @#BKFLAG        ;CLEAR MICROBK TRAP-TO-4 FLAG
15114  056440  076600                  MED                     ;GET FLAG REGISTER
15115  056442  000144                  RDFLAG
15116  056444  052700  100000          BIS     #BIT15,R0       ;SET BIT 15 IN R0
15117  056450  076600                  MED                     ;MED WRITE TO FLAG REG TO
15118  056452  000344          15$:    WRFLAG                  ;ENABLE MICROBK TRAPPING
15119  056454  005000                  CLR     R0              ;IN CASE U-BREAK TRAP DOESN'T OCCOR
15120                                                          ;USUALLY BETTER TO WRITE 0'S
15121  056456  076600                  MED
15122  056460  000000          12$:    .WORD   0
15123  056462  005737  061112          TST     @#BKFLAG        ;DID WE TRAP-TO-4? (FLAG NOT = 0)
15124  056466  001006                  BNE     20$             ;BRANCH IF YES TO NEXT ENTRY
15125  056470  013737  056460  001076  MOV     @#12$,@#$TMP0   ;SAVE MED-CODE FOR ERROR
15126  056476  011237  001100          MOV     (R2),@#$TMP1    ;SAVE EXPECTED U-ADDR FOR ERROR
15127  056502  104015                  ERROR   15              ;MICROBK. TRAP-TO-4 DID NOT OCCUR
15128
15129  056504  105721          20$:    TSTB    (R1)+           ;INCREMENT TO NEXT TABLE
15130  056506  005722                  TST     (R2)+           ;ENTRIES AND
15131  056510  000744                  BR      2$              ;CONTINUE
15132
15133  056512  076600          50$:    MED                     ;GET WHAMI INTO R0
15134  056514  000022                  RDWHAMI
15135  056516  042700  001000          BIC     #BIT9,R0        ; CLEAR BIT 9
15136  056522  076600                  MED                     ;CLEAR THE FLAG REG. TO
15137  056524  000344          13$:    WRFLAG                  ;    DISABLE MICROBK. TRAPPING
15138  056526  076600                  MED                     ;CLEAR THE WHAMI REG. TO
15139  056530  000222          14$:    WRWHAMI                 ;    DISABLE MICROBK. TRAP-TO-4
```

```
15140  056532  012737  056260  001010          MOV     #MEDT11,@#$LPERR        ;RESET LOOP ON ERROR POINTER
15141  056540  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE NORMAL ERROR ROUTINE
15142  056546  012737  000304  177770          MOV     #304,@#UBREAK   ;GENERATE SYNC PULSE ON MED INSTR
15143
15144
15145                                  ;;*****************************************************************
15146                                  ;*TEST 760      PHYSICAL ADDRESS & ODD ADDRESS ERROR LOGGING
15147                                  ;*      THIS TEST CHECKS THAT THE PROPER PHYSICAL ADDRESS BITS
15148                                  ;*      <17:00> ARE LOGGED UPON ERROR.  THE ERROR IS CAUSED BY
15149                                  ;*      FORCING AN ODD ADDRESS TRAP.  THE ERROR LOG MODE USED
15150                                  ;*      IS "LOG FIRST".  ALSO, THE ODD ADDRESS ERROR BITS IN
15151                                  ;*      THE LOG JAM AND CPU ERROR REGISTER ARE CHECKED.
15152                                  ;;*****************************************************************
15153  056554                          TST760:
15154  056554  012700  000757                  MOV     #757,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15155  056560  000004                  SCOPE                           ;CALL THE SCOPE LOOP UTILITY
15156  056562  012737  056622  000004  1$:     MOV     #2$,@#4         ;SETUP PC FOR ODD ADDR SERVICE
15157  056570  012737  000340  000006          MOV     #340,@#6
15158  056576  012700  100001                  MOV     #BIT15+BIT0,R0  ;SETUP "LOG FIRST" MODE
15159  056602  076600                          MED
15160  056604  000222                          WRWHAM1
15161  056606  012702  056563                  MOV     #1$+1,R2        ;SAVE ADDRESS OF ODD ADDR. INSTRUCTION
15162  056612  005767  177745                  TST     1$+1            ;DO ODD ADDRESS INSTRUCTION TO FORCE
15163                                                                  ;A JAMUPP & TRAP TO 4
15164  056616  104023                          ERROR   23              ;*** ODD ADDR. TRAP DID NOT OCCUR
15165  056620  000441                          BR      10$             ;EXIT TEST
15166  056622  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15167  056624  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW
15168  056632  076600                          MED
15169  056634  000100                          RDLJAM
15170  056636  013701  177766                  MOV     @#CPUERR,R1
15171  056642  032701  000100                  BIT     #BIT6,R1          ;WAS ODD ADDR. ERROR RECORDED BY
15172                                                                  ;THE CPU ERROR REGISTER?
15173  056646  001001                          BNE     3$              ;BRANCH IF YES
15174  056650  104024                          ERROR   24              ;*** CPU ERROR REG. DID NOT
15175                                                                  ;REPORT ODD ADDRESS ERROR
15176                                                                  ;READ THE LOG JAM REGISTER
15177  056652  032700  100004          3$:     BIT     #BIT15+BIT2,R0  ;WAS ODD ADDR. ERROR LOGGED BY LOG JAM
15178  056656  001001                          BNE     4$              ;BRANCH IF YES
15179  056660  104024                          ERROR   24              ;*** LOG JAM REG.  DID NOT LOG
15180                                                                  ;ODD ADDRESS ERROR CORRECTLY
15181
15182  056662  005005                  4$:     CLR     R5              ;CLR ERROR FLAG
15183  056664  076600                          MED                     ;READ THE LOG PBA REGISTER
15184  056666  000102                          RDLPBA
15185  056670  010003                          MOV     R0,R3           ;SAVE RECEIVED PHYS ADDR <15:0>
15186  056672  020002                          CMP     R0,R2           ;WERE BITS <15:00> OF THE PHYSICAL
15187                                                                  ;BUS ADDR. LOGGED CORRECTLY?
15188  056674  001401                          BEQ     5$              ;BRANCH IF YES
15189  056676  005205                          INC     R5              ;SET ERROR FLAG
15190  056700  076600                  5$:     MED                     ;READ THE LOG SERVICE REGISTER
15191  056702  000101                          RDLSERVICE
15192  056704  000300                          SWAB    R0              ;GET "PBA 17&16" DOWN TO BIT POSITION 0&1
15193  056706  042700  177774                  BIC     #177774,R0
15194  056712  001002                          BNE     11$             ;BR IF PHYS ADDR BITS <17:16> LOGGED CORRECTLY
15195  056714  005705                          TST     R5              ;PREVIOUS ERROR?
```

```
15196  056716  001402                          BEQ     10$             ;BR IF NOT
15197  056720  005001                  11$:    CLR     R1              ;SET UP EXPECTED PA<17:16>
15198  056722  104026                          ERROR   26              ;*** PHYSICAL BUS ADDR. <17:00>
15199                                                                   ;NOT LOGGED CORRECTLY WHEN
15200                                                                   ;ODD ADDRESS TRAP OCCURRED
15201  056724  005000                  10$:    CLR     R0
15202  056726  076600                          MED                     ;DISABLE "LOG FIRST" MODE
15203  056730  000222                          WRWHAMI
15204
15205
15206                                  ;;*******************************************************************
15207                                  ;*TEST 761       CHECK DISABLE PARITY ERROR TRAP
15208                                          ;*THIS TEST CHECKS THAT PARITY ERROR TRAPS TO LOCATION 114
15209                                          ;*ARE DISABLED WHEN BIT0 OF THE CACHE CONTROL REGISTER IS
15210                                          ;*SET (=1).  A TRAP TO 114 SHOULD NOT OCCUR AND ERROR
15211                                          ;*INFORMATION SHOULD NOT BE LOGGED IN THE LOG PBA, LOG
15212                                          ;*CACHE DATA, OR LOG TAG DATA REGISTERS.  WRONG PARITY IS
15213                                          ;*WRITTEN INTO A TEST LOCATION TO CAUSE THE PARITY ERROR
15214                                          ;*NEEDED IN THIS TEST.
15215                                  ;;*******************************************************************
15216  056732                          TST761:
15217  056732  012700  000760                  MOV     #760,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15218  056736  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15219
15220  056740  012701  064046                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15221  056744  005711                          TST     (R1)            ;MAKE IT A HIT
15222  056746  012737  000100  177746          MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15223  056754  012711  125252                  MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15224  056760  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE PARITY ERROR TRAPS
15225                                                                   ;AND CLEAR WWP
15226  056766  012737  057026  000114          MOV     #1$,@#114       ;SETUP PARITY ERROR VECTOR
15227  056774  012737  000340  000116          MOV     #340,@#116
15228  057002  005000                          CLR     R0
15229  057004  076600                          MED                     ;CLEAR LOG PBA REGISTER
15230  057006  000302                          WRLPBA
15231  057010  076600                          MED                     ;CLEAR LOG CACHE DATA REGISTER
15232  057012  000306                          WRLDATA
15233  057014  076600                          MED                     ;CLEAR LOG CACHE TAG REGISTER
15234  057016  000307                          WRLTAG
15235  057020  005767  005022                  TST     TLOC1           ;READ TEST LOC0 TO FORCE PARITY ERROR
15236  057024  000406                          BR      2$              ;BRANCH IF NO TRAP OCCURS
15237  057026  012700  000200          1$:     MOV     #200,R0
15238  057032  076600                          MED                     ;CLEAN UP THE CACHE
15239  057034  000352                          352                     ;INITIALIZATION CODE
15240  057036  022626                          CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15241  057040  104030                          ERROR   30              ;*** PARITY TRAP TO 114 OCCURRED
15242                                                                   ;WHEN IT SHOULD HAVE BEEN DISABLED
15243  057042  012700  000200          2$:     MOV     #200,R0
15244  057046  076600                          MED                     ;CLEAN UP THE CACHE
15245  057050  000352                          352                     ;INITIALIZATION CODE
15246  057052  012711  125252                  MOV     #125252,(R1)    ;WRITE BAK GOOD PARITY IN TST LOC
15247  057056  012737  000116  000114          MOV     #116,@#114      ;RESTORE ORIGINAL PARITY HANDLER & PSW
15248  057064  005037  000116                  CLR     @#116
15249  057070  005005                          CLR     R5              ;CLEAR ERROR FLAG
15250  057072  076600                          MED                     ;READ LOG PBA REGISTER
15251  057074  000102                          RDLPBA
```

```
15252  057076  010003                    MOV     R0,R3           ;SAVE COPY
15253                                                            ;LOG PBA REG. STILL CLEAR?
15254  057100  001401                    BEQ     3$              ;BRANCH IF YES
15255  057102  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15256  057104  076600            3$:     MED                     ;READ LOG CACHE DATA REG.
15257  057106  000106                    RDLDATA
15258  057110  010001                    MOV     R0,R1           ;SAVE COPY
15259                                                            ;LOG CACHE DATA REG. STILL CLEAR?
15260  057112  001401                    BEQ     4$              ;BRANCH IF YES
15261  057114  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15262  057116  076600            4$:     MED                     ;READ LOG CACHE TAG REG.
15263  057120  000107                    RDLTAG
15264  057122  010002                    MOV     R0,R2           ;SAVE COPY
15265                                                            ;LOG CACHE TAC REG. STILL CLEAR?
15266  057124  001401                    BEQ     5$              ;BRANCH IF YES
15267  057126  005205                    INC     R5              ;OTHERWISE SET ERROR FLAG
15268  057130  005705            5$:     TST     R5              ;WERE ANY OF LOG REGISTERS CHANGED
15269  057132  001401                    BEQ     6$              ;BRANCH IF NO
15270  057134  104027                    ERROR   27              ;*** ONE OF LOG REGISTERS CHANGED
15271                                                            ;WHEN ERROR SHOULD NOT HAVE BEEN LOGGED
15272                                                            ;LOG PBA, LOG DATA & LOG TAG
15273                                                            ;REGISTER SHOULD BE CLEAR.
15274  057136  005037  177746    6$:     CLR     @#CCR           ;ENABLE PARITY ERROR TRAPS
15275
15276                            ;;*****************************************************************
15277                            ;*TEST 762       CHECK PARITY ERROR BITS IN MEMERR REG. IN BACKUP MODE OF CACHE (TRAP)
15278
15279                                    ;*THIS TEST CHECKS THAT ALL OF THE PARITY ERROR BITS (5,6,7)
15280                                    ;*OF THE MEMORY ERROR REGISTER ARE SET TO "1" WHEN A CACHE
15281                                    ;*PARITY ERROR OCCURS IN THE BACKUP MODE.
15282                            ;;*****************************************************************
15283  057142                    TST762:
15284  057142  012700  000761            MOV     #761,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15285  057146  000004                    SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15286  057150  012701  064046            MOV     #TLOC1,R1       ;GET POINTER TO TEST LOCATION
15287  057154  005711                    TST     (R1)            ;MAKE IT A HIT
15288  057156  012737  000100  177746    MOV     #WWP,@#CCR      ;SET WRITE WRONG PARITY BIT
15289  057164  012711  125252            MOV     #125252,(R1)    ;WRITE TO TEST LOC. WITH WRONG PARITY
15290  057170  042737  000100  177746    BIC     #WWP,@#CCR      ;CLEAR WWP
15291  057176  012737  057232  000114    MOV     #1$,@#114       ;SETUP NEW TEST HANDLER AT PARITY VECTOR
15292  057204  012737  000340  000116    MOV     #340,@#116
15293  057212  005737  064046            TST     @#TLOC1         ;READ TEST LOC. TO FORCE PARITY ERROR
15294  057216  012700  000200            MOV     #200,R0
15295  057222  076600                    MED                     ;CLEAN UP THE CACHE
15296  057224  000352                    352                     ;INITIALIZATION CODE
15297  057226  104031                    ERROR   31              ;*** PARITY ERROR DID NOT CAUSE TRAP
15298  057230  000405                    BR      2$              ;BRANCH TO 2$
15299  057232  012700  000200    1$:     MOV     #200,R0
15300  057236  076600                    MED                     ;CLEAN UP THE CACHE
15301  057240  000352                    352                     ;INITIALIZATION CODE
15302  057242  022626                    CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15303  057244  022737  000340  177744    CMP     #000340,@#MEMERR  ;WERE PARITY ERROR BITS (5,6,7) SET
15304                                                            ;AND CPU ABORT BIT (15) LEFT CLEAR
15305                                                            ;IN MEMORY ERROR REGISTER?
15306  057252  001403                    BEQ     3$              ;BRANCH IF YES
15307  057254  013700  177744            MOV     @#MEMERR,R0
```

```
 15308  057260  104032                          ERROR   32              ;*** MEMORY ERROR REGISTER BITS
 15309                                                                   ;WERE SET INCORRECTLY
 15310  057262  012737  000116  000114  3$:     MOV     #116,@#114      ;RESTORE OLD PARITY HANDLER PC & PSW
 15311  057270  005037  000116                  CLR     @#116
 15312
 15313                                  ;;*****************************************************************
 15314                                  ;*TEST 763      CHECK UNIBUS TIMEOUT, ODD ADDRESS AND LOG CONTINUOUS MODE
 15315
 15316                                  ;*THIS TEST CHECKS THAT THE "UNIBUS TIMEOUT" BIT (BIT4)
 15317                                  ;*GETS SET IN THE CPU ERROR REGISTER WHEN A TIMEOUT OCCURS.
 15318                                  ;*A TIMEOUT TRAP IS FORCED BY REFERENCING BUS ADDRESS 760000.
 15319                                  ;*THEN AN ODD ADDRESS ERROR IS FORCED AND IT
 15320                                  ;*IS CHECKED IF ONLY BIT (6)-ODD ADDRESS ERROR IS SET
 15321                                  ;*(IN CPUERR).  THIS CHECKS THAT THE ERROR LOG IS
 15322                                  ;*CONTINUOSLY UPDATED IN THE "LOG CONTINUOUS" MODE.
 15323                                  ;;*****************************************************************
 15324  057274                          TST763:
 15325  057274  012700  000762                  MOV     #762,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
 15326  057300  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
 15327  057302  012737  057324  000004          MOV     #1$,@#4         ;SETUP NEW PC & PSW FOR THE
 15328  057310  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT SERVICE ROUTINE
 15329  057316  005737  160000                  TST     @#160000        ;FORCE A TIMEOUT TRAP TO 4 BY
 15330                                                                   ;REFERENCING NON-EXISTENT ADDRESS
 15331  057322  000461                          BR      6$
 15332  057324  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
 15333  057326  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD PC & PSW FOR TIMEOUT
 15334  057334  076600                          MED
 15335  057336  000100                          RDLJAM
 15336  057340  013701  177766                  MOV     @#CPUERR,R1     ;SAVE CPU ERR REG
 15337  057344  022701  000020                  CMP     #BIT4,R1        ;DID "UNIBUS TIMEOUT" BIT IN CPU ERROR
 15338                                                                   ;REGISTER GET SET?
 15339  057350  001401                          BEQ     2$              ;BRANCH IF YES
 15340  057352  104033                          ERROR   33              ;*** "UNIBUS TIMEOUT" BIT (BIT4) IN CPU
 15341                                                                   ;ERROR REG. DID NOT SET WHEN A
 15342                                                                   ;TIMEOUT WAS FORCED
 15343                                                                   ;READ THE LOG JAM REGISTER
 15344  057354  022700  021200          2$:     CMP     #BIT13+BIT9+BIT7,R0     ;                              (REV.D0)
 15345                                                                   ;DID "UNIBUS TIMEOUT" BIT (BIT7) SET?
 15346  057360  001401                          BEQ     3$              ;BRANCH IF YES
 15347  057362  104033                          ERROR   33              ;*** "UNIBUS TIMEOUT" BIT (BIT7)
 15348                                                                   ;DID NOT SET IN LOG JAM REGISTER
 15349                                                                   ;WHEN UNIBUS TIMEOUT WAS FORCED
 15350  057364  076600                  3$:     MED                     ;READ LOG PBA
 15351  057366  000102                          RDLPBA
 15352  057370  020027  160000                  CMP     R0,#160000      ;WAS PHYS BA LOGGED CORRECTLY?
 15353  057374  001403                          BEQ     5$
 15354  057376  012701  160000                  MOV     #160000,R1
 15355  057402  104020                          ERROR   20              ;PHYSICAL BUS ADDRESS WAS
 15356                                                                   ;LOGGED WRONG ON A UNIBUS
 15357                                                                   ;TIMEOUT
 15358  057404  012737  057426  000004  5$:     MOV     #4$,@#4         ;SET UP PC,PSW FOR ODD ADDRESS
 15359  057412  012737  000340  000006          MOV     #340,@#6
 15360  057420  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS ERROR
 15361  057424  000420                          BR      6$
 15362  057426  022626                  4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
 15363  057430  012737  061220  000004          MOV     #BERR,@#4
```

```
15364  057436  076600                          MED
15365  057440  000100                          RDLJAM
15366  057442  013701  177766                  MOV     @#CPUERR,R1
15367  057446  022701  000100                  CMP     #BIT6,R1        ;ODD ADDR. BUT SET 3
15368  057452  001401                          BEQ     7$
15369  057454  104024                          ERROR   24              ;ODD ADDRESS BIT WAS
15370                                                                  ;NOT SET IN THE CPU
15371                                                                  ;ERROR REGISTER.  IN LOG
15372                                                                  ;CONTINUOUS MADE THE
15373                                                                  ;LATEST ERROR SHOULD
15374                                                                  ;BE LOGGED
15375  057456  032700  000004          7$:     BIT     #BIT2,R0        ;ODD ADR. BIT SET IN
15376  057462  001001                          BNE     6$              ;LOG JAM?
15377  057464  104024                          ERROR   24              ;ODD ADDRESS BIT WAS
15378                                                                  ;NOT SET IN THE LOG
15379                                                                  ;JAM REGISTER ON A
15380                                                                  ;ODD ADDRESS ERROR
15381  057466  076600                  6$:     MED                     ;CHECK IF LAST INTERRUPT VECTOR
15382  057470  000104                          RDLFGINT                ;WAS LOGGED?
15383  057472  120027  000004                  CMPB    R0,#4
15384  057476  001401                          BEQ     8$
15385  057500  104036                          ERROR   36              ;LAST ERROR VECTOPR WS NOT LOGGED
15386
15387  057502                          8$:
15388
15389
15390                                  ;;*******************************************************************
15391                                  ;*TEST 764       CHECK ILLEGAL INTERNAL ADDRESS TRAP
15392
15393                                          ;*THIS TEST CHECKS THAT A TRAP OCCURS UPON REFERENCING AN
15394                                          ;*ILLEGAL INTERNAL ADDRESS AND THAT "ILLEGAL INTERNAL ADDRESS"
15395                                          ;*BIT (BIT0) OF THE CPU ERROR REGISTER AND BIT5 OF LOG JAM
15396                                          ;*REGISTER GET SET.  IT ALSO CHECKS IF THE INTERRUPT VECTOR
15397                                          ;*(4) IS SAVED AS THE "LAST INTERRUPT VECTOR" IN THE LOG
15398                                          ;*FLAG/INTERRUPT REG.
15399                                  ;;*******************************************************************
15400  057502                          TST764:
15401  057502  012700  000763                  MOV     #763,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15402  057506  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15403  057510  012737  057540  000004          MOV     #1$,@#4         ;SETUP NEW HANDLER PC & PSW
15404  057516  012737  000340  000006          MOV     #340,@#6
15405  057524  005037  177746                  CLR     @#CCR
15406  057530  012707  177746                  MOV     #CCR,PC         ;ILLEGAL INTERNAL ADDRESS TRAP SHOULD OCCUR
15407  057534  104034                          ERROR   34              ;*** ILLEGAL INTERNAL ADDRESS
15408                                                                  ;DID NOT RESULT IN A TRAP
15409  057536  000420                          BR      3$              ;BRANCH TO EXIT IF NO TRAP
15410  057540  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15411  057542  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE OLD HANDLER PC & PSW
15412  057550  076600                          MED
15413  057552  000100                          RDLJAM
15414  057554  013701  177766                  MOV     @#CPUERR,R1
15415  057560  032701  000001                  BIT     #BIT0,R1        ;DID "ILLEGAL INTERNAL ADDRESS" BIT (0)
15416                                                                  ;IN CPU ERROR REGISTER GET SET?
15417  057564  001001                          BNE     2$              ;BRANCH IF YES
15418  057566  104035                          ERROR   35              ;*** ILLEGAL INTERNAL ADDRESS
15419                                                                  ;BIT DID NOT SET IN CPU ERROR REG.
```

```
15420
15421  057570  032700  000040          2$:     BIT     #BIT5,R0        ;READ THE LOG JAM REG.
15422                                                                   ;DID "ILLEGAL INTERNAL ADDRESS" BIT (5)
15423  057574  001001                          BNE     3$              ;IN LOG JAM REG. GET SET
15424  057576  104055                          ERROR   35              ;BRANCH IF YES
15425                                                                   ;*** ILLEGAL INTERNAL ADDRESS BIT
15426  057600                          3$:                             ;DID NOT SET IN LOG JAM REG.
15427
15428
15429                                  ;;*****************************************************************
15430                                  ;*TEST 765      CHECK LOG SERVICE & MEMERR LOGS LO-HI BYTE & TAG, IN CACHE ABORT MODE
15431                                          ;TEST CHECKS THAT "LO BYTE PARITY" "HI BYTE PARITY" AND "TAG PARITY"
15432                                          ;BITS CAN SET IN "LOG SERVICE" REGISTERS.  IT IS ALSO
15433                                          ;CHECKED THAT THE PROPER TAG AND DATA BITS GET STORED
15434                                          ;IN THE "LOG CACHE DATA, "LOG CACHE TAG/CPU" AND THE
15435                                          ;"MEMORY ADDRESS REGISTER" WHEN A PARITY ERROR IS
15436                                          ;FORCED.
15437                                          ;IT IS CHECKED IF THE INSTRUCTION WAS ABORTED AND THE
15438                                          ;LOG FLAG/INTERRUPT REGISTER LOGGED THE LAST INTERRUPT
15439                                          ;VECTOR.
15440                                  ;;*****************************************************************
15441  057600                          TST765:
15442  057600  012700  000764                  MOV     #764,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15443  057604  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15444
15445  057606  012737  000201  177746          MOV     #DPTRP+PABORT,@#CCR  ;DISABLE PARITY TRAPS (CACHE)
15446  057614  005037  001062                  CLR     @#$REG0
15447  057620  012701  064046                  MOV     #TLOC1,R1       ;GET POINTER TO TEST LOC.
15448  057624  005711                          TST     (R1)            ;MAKE IT A HIT
15449  057626  052737  000100  177746          BIS     #WWP,@#CCR      ;WRITE WRONG PARITY SET
15450  057634  012711  125252                  MOV     #125252,(R1)    ;WRITE TEST LOCATION WITH WRONG PARITY
15451  057640  042737  000100  177746          BIC     #WWP,@#CCR      ;CLEAR WWP
15452  057646  012700  100001                  MOV     #BIT15+BIT0,R0
15453  057652  076600                          MED                     ;ENABLE "LOG FIRST" MODE, AND
15454  057654  000222                          WRWHAMI                 ;ERROR LOGGING
15455  057656  042737  000001  177746          BIC     #DPTRP,@#CCR    ;ENABLE CACHE PARITY TRAPS
15456  057664  012737  057712  000114          MOV     #PTRP1,@#114    ;NEW PARITY TRAP SERVICE
15457  057672  016737  004150  001062          MOV     TLOC1,@#$REG0   ;READ TEST LOC, FORCE PARITY ERROR
15458  057700  012700  000200                  MOV     #200,R0
15459  057704  076600                          MED                     ;CLEAN UP THE CACHE
15460  057706  000352                          352                     ;INITIALIZATION CODE
15461  057710  104031                          ERROR   31              ;*** CACHE PARITY ERROR TRAP
15462                                                                   ;DID NOT OCCUR WHEN
15463                                                                   ;TEST LOC WITH BAD PARITY
15464                                                                   ;WAS READ
15465
15466                                                                   ;ENTER HERE IF PARITY TRAP OCCURRED
15467  057712  012700  000200          PTRP1:  MOV     #200,R0
15468  057716  076600                          MED                     ;CLEAN UP THE CACHE
15469  057720  000352                          352                     ;INITIALIZATION CODE
15470  057722  012737  000001  177746          MOV     #DPTRP,@#CCR    ;DISABLE CACHE PARITY ERROR TRAPS
15471  057730  012737  000116  000114          MOV     #116,@#114      ;REESTABLISH OLD SERVICE VECTORS
15472  057736  005037  000116                  CLR     @#116
15473  057742  022626                          CMP     (SP)+,(SP)+
15474  057744  005737  001062                  TST     @#$REG0         ;WAS THE INSTRUCTION ABORTED ON
15475                                                                   ;CACHE PARITY ERROR (ABORT MODE)?
```

```
15476  057750  001401                  BEQ     1$              ;YES
15477  057752  104041                  ERROR   41              ;INSTRUCTION HAVING CACHE PARITY
15478                                                          ;ERROR WAS NOT ABORTED, IN THE
15479                                                          ;CACHE ABORT MODE.
15480  057754  076600          1$:     MED                     ;READ THE "LOG SERVICE" REGISTER
15481  057756  000101                  RDLSERVICE
15482  057760  010004                  MOV     R0,R4           ;COPY
15483  057762  042704  177435          BIC     #^C<LO+HI+TAG+BIT1>,R4  ;MASK ALL BUT LO,HI,TAG BITS
15484  057766  022704  000342          CMP     #342,R4         ;LO,HI ,TAG, CACHE PARITY BITS SET? IN "SERVICE"
15485  057772  001401                  BEQ     2$              ;YES
15486  057774  104042                  ERROR   42              ;*** "LO BYTE" PARITY ERROR
15487                                                          ;AND "TAG" PARITY ERROR BITS
15488                                                          ;WERE NOT LOGGED CORRECTLY IN "LOG
15489                                                          ;SERVICE" REGISTER, WHEN PARITY
15490                                                          ;ERROR TRAP WAS FORCED.
15491                                                          ;CLEAR BITS ARE ACTIVE.
15492  057776  013700  177744  2$:     MOV     @#MEMERR,R0     ;GET MEM ERR REG
15493  060002  022700  100340          CMP     #HI+LO+TAG+BIT15,R0  ;DID"LO BYTE" "HI BYTE" AND "TAG"
15494                                                          ;PARITY ERROR BITS SET IN
15495                                                          ;THE MEMORY ERROR REGISTER?
15496  060006  001401                  BEQ     3$              ;YES
15497  060010  104043                  ERROR   43              ;*** "LO BYTE" "HI BYTE" AND "TAG" PARITY
15498                                                          ;ERROR BITS DID NOT SET
15499                                                          ;CORRECTLY IN THE MEMORY
15500                                                          ;ERROR REGISTER
15501  060012  076600          3$:     MED                     ;READ "LOG PBA" REGISTER
15502  060014  000102                  RDLPBA
15503  060016  020027  064046          CMP     R0,#TLOC1       ;DID "LOG PBA" CONTAIN CORRECT
15504                                                          ;PHYSICAL BUS ADDRESS-WHERE
15505                                                          ;THE PARITY ERROR OCCURRED?
15506  060022  001403                  BEQ     4$              ;YES
15507  060024  012701  064046          MOV     #TLOC1,R1       ;EXPECTED PBA
15508  060030  104020                  ERROR   20              ;*** PHYSICAL BUS ADDRESS
15509                                                          ;(WHERE PARITY ERROR OCCURRED)
15510                                                          ;WAS NOT LOGGED CORRECTLY
15511                                                          ;WHEN CACHE PARITY ERROR WAS FORCED
15512  060032  076600          4$:     MED
15513  060034  000107                  RDLTAG                  ;READ "LOG CACHE TAG" REGISTER
15514  060036  000300                  SWAB    R0
15515  060040  012701  064046          MOV     #TLOC1,R1       ;SHIFT RIGHT (3 TIMES) THE 16 BIT
15516  060044  000301                  SWAB    R1
15517  060046  106201                  ASRB    R1              ;PHYSICAL BUS ADDRESS OF THE
15518  060050  106201                  ASRB    R1              ;TEST LOCATION
15519  060052  106201                  ASRB    R1
15520  060054  052701  000200          BIS     #BIT7,R1        ;FUDGE TAGE BIT
15521  060060  120100                  CMPB    R1,R0           ;WAS THE CORRECT TAG LOGGED?
15522  060062  001401                  BEQ     5$              ;YES
15523  060064  104017                  ERROR   17              ;TAG BITS WERE NOT LOGGED
15524                                                          ;CORRECTLY, WHEN CACHE
15525                                                          ;PARITY ERROR WAS FORCED
15526  060066  076600          5$:     MED                     ;READ CACHE DATA
15527  060070  000106                  RDLDATA
15528  060072  020027  125252          CMP     R0,#125252      ;CACHE DATA LOGGED CORRECTLY?
15529  060076  001403                  BEQ     6$
15530  060100  012701  125252          MOV     #125252,R1      ;EXPECTED DATA
15531  060104  104016                  ERROR   16
```

```
15532
15533  060106  012700  000001          6$:     MOV     #BIT0,R0        ;SET UP LOG CONTINUOUS
15534  060112  076600                          MED
15535  060114  000222                          WRWHAMI
15536  060116  012737  060130  000004          MOV     #7$,@#4         ;SETUP CPU VECTOR
15537  060124  005737  160000                  TST     @#160000        ;FORCE TIMEOUT & TRAP TO 7$
15538  060130  022626                  7$:     CMP     (SP)+,(SP)+
15539  060132  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE CPU VECTOR
15540  060140  076600                          MED                     ;READ LOG FLAG/INTERRUPT REGISTER
15541  060142  000104                          RDLFGINT
15542  060144  120027  000114                  CMPB    R0,#114         ;DID LO BYTE CONTAIN VECTOR 114?
15543  060150  001403                          BEQ     8$
15544  060152  010037  001062                  MOV     R0,@#$REG0
15545  060156  104036                          ERROR   36              ;LAST INTERRUPT VECTOR WAS NOT
15546                                                                  ;LOGGED CORRECTLY IN FLAG REGISTER
15547                                                                  ;WHEN A CACHE PARITY ERROR WAS
15548                                                                  ;FORCED.
15549  060160                          8$:
15550
15551                                   ;;*****************************************************************
15552                                   ;*TEST 766       CHECK "LOG FIRST" MODE OF ERROP LOGGING
15553                                           ;*THIS TEST CHECKS THE "LOG FIRST" MODE OF ERROR LOGGING.
15554                                           ;*THE "LOG FIRST" MODE IS ENABLED.  THEN A TIME-OUT TRAP
15555                                           ;*IS FORCED, BIT 4 OF CPU ERROR REGISTER SHOULD BE SET.
15556                                           ;*THEN AN ODD ADDRESS TRAP IS FORCED.  HOWEVER, THIS
15557                                           ;*TIME THE ERROR SHOULD NOT BE LOGGED; BIT 6 (ODD
15558                                           ;*ADDRESS) SHOULD NOT BE SET BECAUSE THE ERROR LOG
15559                                           ;*IS LOCKED UP AFTER THE FIRST ERROR.
15560
15561                                           ;*THEN, THE ERROR LOG IS ENABLED (BY SETTING BIT 0 OF
15562                                           ;*WHAMI).  AN ODD ADDRESS ERROR IS FORCED AGAIN AND IT IS
15563                                           ;*CHECKED THAT THIS TIME THE ERROR IS LOGGED, (BIT 6-ODD
15564                                           ;*ADDRESS SHOULD BE SET IN CPU ERROR REGISTER).
15565                                   ;;*****************************************************************
15566  060160                          TST766:
15567  060160  012700  000765                  MOV     #765,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SCOPE
15568  060164  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
15569
15570  060166  012700  100001                  MOV     #BIT15+BIT0,R0  ;SET UP "LOG FIRST MODE
15571  060172  076600                          MED
15572  060174  000222                          WRWHAMI
15573  060176  012737  060220  000004          MOV     #1$,@#4         ;SET UP NEW PC & PSW FOR
15574  060204  012737  000340  000006          MOV     #340,@#6        ;TIMEOUT
15575  060212  005737  160C00                  TST     @#160000        ;FORCE A TIMEOUT
15576  060216  000462                          BR      5$              ;SKIP TEST IF NO TIMEOUT
15577
15578  060220  022626                  1$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15579                                                                  ;BIT 4 OF CPU ERROR REGISTER
15580                                                                  ;SHOULD HAVE SET
15581  060222  012737  060236  000004          MOV     #2$,@#4         ;SET UP NEW PC FOR ODD ADDRESS
15582  060230  005767  177765                  TST     1$+1            ;FORCE ODD ADDRESS TRAP
15583  060234  000453                          BR      5$              ;SKIP TEST IF NO ODD ADDRESS TRAP
15584
15585  060236  022626                  2$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15586  060240  012737  061220  000004          MOV     #BERR,@#4
15587  060246  076600                          MED
```

```
15588  060250  000100                          RDLJAM
15589  060252  013701  177766                  MOV     @#CPUERR,R1
15590  060256  022701  000020                  CMP     #BIT4,R1        ;"TIMEOUT" BIT SHOULD BE STILL
15591                                                                  ;SET, CHECK?
15592  060262  001402                          BEQ     3$
15593  060264  104033                          ERROR   33              ;*** SECOND ERROR (ODD ADDRESS)
15594                                                                  ;UPDATED THE ERROR LOG IN
15595                                                                  ;THE LOG FIRST MODE.  BIT 4
15596                                                                  ;(UNIBUS TIMEOUT) SHOULD BE
15597                                                                  ;STILL SET FROM THE FIRST
15598                                                                  ;ERROR
15599  060266  000436                          BR      5$              ;SKIP THE REST
15600  060270  032700  100004          3$:     BIT     #BIT2+BIT15,R0  ;CHECK THAT ODD ADRES ERROR BITS NOT
15601  060274  001401                          BEQ     6$              ;SET IN LOG JAM.  NOTE LOG FIRST
15602                                                                  ;MODE SHOULD INHIBIT FURTHER
15603                                                                  ;ERROR LOGGING
15604  060276  104037                          ERROR   37              ;ODD ADRESS ERROR BITS GOT SET IN LOG JAM
15605                                                                  ;THEY SHOULD NOT BE SINCE LOG FIRST MODE
15606                                                                  ;INHIBITS ERROR LOGGING AFTER THE FIRST ERROR
15607  060300  012700  100001          6$:     MOV     #BIT15+BIT0,R0  ;ENABLE ERROR LOG AGAIN IN
15608                                                                  ;LOG FIRST MODE
15609  060304  076600                          MED
15610  060306  000222                          WRWHAMI
15611  060310  012737  060332  000004          MOV     #4$,@#4         ;SET UP NEW PC & PSW FOR
15612  060316  012737  000340  000006          MOV     #340,@#6        ;ODD ADDRESS ERROR
15613  060324  005767  177741                  TST     3$+1            ;FORCE ODD ADDRESS TRAP
15614  060330  000415                          BR      5$              ;SKIP IF NO TRAP
15615  060332  022626                  4$:     CMP     (SP)+,(SP)+     ;RESTORE STACK
15616                                                                  ;RESTORE OLD PC(4), PSW(6)
15617  060334  012737  061220  000004          MOV     #BERR,@#4
15618  060342  022737  000100  177766          CMP     #BIT6,@#CPUERR  ;THE ERROR LOG FROM PREVIOUS
15619                                                                  ;ERROR SHOULD BE OVER WRITTEN.
15620                                                                  ;ODD ADDRESS BIT SHOULD
15621                                                                  ;BE SET, BECAUSE THE ERROR
15622  060350  001405                          BEQ     5$              ;LOG WAS ENABLED.
15623                                                                  ;OK, IF YES
15624  060352  076600                          MED
15625  060354  000100                          RDLJAM
15626  060356  013701  177766                  MOV     @#CPUERR,R1
15627  060362  104040                          ERROR   40              ;THE ERROR LOG WAS NOT UPDATED
15628                                                                  ;(UPON AN ODD ADDRESS ERROR)
15629                                                                  ;AFTER THE LOG WAS ENABLED.
15630                                                                  ;AT THIS FORMAT BIT 6 OF
15631                                                                  ;CPU ERROR REGISTER SHOULD
15632                                                                  ;BE SET.  IT WAS NOT.
15633  060364  012737  06122C  000004  5$:     MOV     #BERR,@#4       ;RESTORE OLD PC(4), PSW(6)
15634  060372  012700  000001                  MOV     #BIT0,R0
15635  060376  076600                          MED                     ;PUT THE LOGGING BACK INTO
15636  060400  000222                          WRWHAMI                 ;"CONTINUOUS" MODE
15637
15638                                  ;;*******************************************************************
15639                                  ;*TEST 767       CHECK LAST INTRRUPT VECTOR IS LOGGED IN FLAG REG.
15640                                  ;;*******************************************************************
15641  060402                          TST767:
15642  060402  012700  000766                  MOV     #766,R0         ;;SETUP MISSED TEST & FULL WRD TEST # PRIOR TO SLOPE
15643  060406  000004                          SCOPE                   ;CALL THE SCOPE LOOP UTILITY
```

```
15644
15645  060410  012737  060420  000030          MOV     #1$,@#30        ;LOAD EMT VECTOR WITH 1$
15646  060416  104000                          EMT                     ;FIRST INTERRUPT -- EMT
15647  060420  022626                  1$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15648  060422  012737  061620  000030          MOV     #$ERROR,@#30    ;RESTORE VECTOR
15649  060430  012737  060442  000004          MOV     #2$,@#4         ;SET UP CPU VECTOR
15650  060436  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15651  060442  022626                  2$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15652  060444  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15653  060452  076600                          MED                     ;CHECK FLAG
15654  060454  000104                          RDLFGINT
15655  060456  120027  000030                  CMPB    R0,#30          ;EMT VECTOR LAST LOGGED?
15656  060462  001401                          BEQ     3$              ;BR IF YES
15657  060464  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15658                                                                  ;LO BYTE OF LOG FLAG/INT REG S/B=30
15659
15660  060466  012737  060476  000020  3$:     MOV     #4$,@#20        ;LOAD IOT VECTOR WITH 4$
15661  060474  000004                          IOT                     ;SECOND INTERRUPT-SHOULD LOAD LOG FLAG REG
15662  060476  022626                  4$:     CMP     (SP)+,(SP)+     ;CLEANUP STACK
15663  060500  012737  061260  000020          MOV     #$SCOPE,@#20    ;RESTORE IOT VECTOR
15664  060506  012737  060520  000004          MOV     #5$,@#4         ;SET UP CPU VECTOR
15665  060514  005737  160000                  TST     @#160000        ;FORCE TIMEOUT
15666  060520  022626                  5$:     CMP     (SP)+,(SP)+     ;CLEAN UP STACK
15667  060522  012737  061220  000004          MOV     #BERR,@#4       ;RESTORE BUS ERROR VECTOR
15668  060530  076600                          MED                     ;CHECK FLAG
15669  060532  000104                          RDLFGINT
15670  060534  120027  000020                  CMPB    R0,#20          ;IOT VECTOR LAST LOGGED?
15671  060540  001401                          BEQ     6$              ;BR IF YES
15672  060542  104036                          ERROR   36              ;LOG FLAG/INT REG DID NOT LOG VECTOR
15673                                                                  ;LOW BYTE S/B = 20
15674
15675  060544  012700  000767          6$:     MOV     #$TN-1,R0       ;SET UP FOR MISSED TEST CHECK AND
15676                                                                  ;FULL WORD TEST NUMBER FOR APT
15677
15678                                  .ENABLE AMA
15679
15680
15681                                  .SBTTL  END OF PASS ROUTINE
15682
15683                                  ;;**********************************************************************
15684                                  ;*INCREMENT THE PASS NUMBER ($PASS)
15685                                  ;*IF THERES A MONITOR GO TO IT
15686                                  ;*IF THERE ISN'T JUMP TO INIT
15687
15688  060550                          $EOP:
15689  060550  000004                          SCOPE
15690  060552  005037  001110                  CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
15691  060556  005237  001126                  INC     $PASS           ;;INCREMENT THE PASS NUMBER
15692  060562  042737  100000  001126          BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
15693  060570  005327                          DEC     (PC)+           ;;LOOP?
15694  060572  000001                  $EOPCT: .WORD   1
15695  060574  003027                          BGT     $DOAGN          ;;YES
15696  060576  012737                          MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
15697  060600  000001                  $ENDCT: .WORD   1
15698  060602  060572                          $EOPCT
15699  060604  104401  065103                  TYPE    ,EOP1           ;TYPE "END PASS #"
```

```
15700  060610  013746  001126              MOV     $PASS,-(SP)     ;SAVE $PASS FOR TYPOUT
15701  060614  104402                      TYPOC                   ;TYPE PASS NUMBER IN OCTAL
15702  060616  104401  065121              TYPE    ,EOP2           ;TYPE "ERROR COUNT ="
15703  060622  013746  001012              MOV     $ERTTL,-(SP)    ;SAVE ERROR TOTAL FOR TYPOUT
15704  060626  104402                      TYPOC                   ;TYPE ERROR TOTAL
15705  060630  104401  001115              TYPE    ,$CRLF
15706  060634  013700  000042      $GET42: MOV     @#42,R0         ;;GET MONITOR ADDRESS
15707  060640  001405                      BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
15708  060642  000005                      RESET                   ;;CLEAR THE WORLD
15709  060644  004710              $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
15710  060646  000240                      NOP                     ;;SAVE ROOM
15711  060650  000240                      NOP                     ;;FOR
15712  060652  000240                      NOP                     ;;ACT11
15713  060654                      $DOAGN:
15714  060654  000137                      JMP     @(PC)+          ;;RETURN
15715  060656  003262              $RTNAD: .WORD   INIT
15716  060660     377      377     000  $ENULL: .BYTE   -1,-1,0         ;;NULL CHARACTER STRING
15717          060664                      .EVEN
```

```
15718                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15719                                           .SBTTL  /  /  /  /  /  /  UTILITIES  /  /  /  /  /  /
15720                                   ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
15721
15722                                   .SBTTL  POWER DOWN AND UP ROUTINES
15723
15724                                   ;;*****************************************************************
15725                                   ;POWER DOWN ROUTINE
15726  060664  012737  061036  000024  $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
15727  060672  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15728  060700  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
15729  060702  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
15730  060704  010246                          MOV     R2,-(SP)        ;;PUSH R2 ON STACK
15731  060706  010346                          MOV     R3,-(SP)        ;;PUSH R3 ON STACK
15732  060710  010446                          MOV     R4,-(SP)        ;;PUSH R4 ON STACK
15733  060712  010546                          MOV     R5,-(SP)        ;;PUSH R5 ON STACK
15734  060714  017746  120120                  MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
15735  060720  010637  061042                  MOV     SP,$SAVR6       ;;SAVE SP
15736  060724  012737  060736  000024          MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
15737  060732  000000                          HALT
15738  060734  000776                          BR      .-2             ;;HANG UP
15739
15740                                   ;;*****************************************************************
15741                                   ;POWER UP ROUTINE
15742  060736  012737  061036  000024  $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
15743  060744  013706  061042                  MOV     $SAVR6,SP       ;;GET SP
15744  060750  005037  061042                  CLR     $SAVR6          ;;WAIT LOOP FOR THE TTY
15745  060754  005237  061042          1$:     INC     $SAVR6          ;;WAIT FOR THE INC
15746  060760  001375                          BNE     1$              ;;OF  WORD
15747  060762  011600                          MOV     (SP),R0         ;GET OLD SWR VALUE
15748  060764  076600                          MED                     ;WRITE BACK ORIGINAL SWR VALUE
15749  060766  000226                          WCNSSW                  ;INTO HARDWARE SWITCH REGISTER
15750  060770  012677  120044                  MOV     (SP)+,@SWR      ;;POP STACK INTO @SWR
15751  060774  012605                          MOV     (SP)+,R5        ;;POP STACK INTO R5
15752  060776  012604                          MOV     (SP)+,R4        ;;POP STACK INTO R4
15753  061000  012603                          MOV     (SP)+,R3        ;;POP STACK INTO R3
15754  061002  012602                          MOV     (SP)+,R2        ;;POP STACK INTO R2
15755  061004  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
15756  061006  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
15757  061010  012737  060664  000024          MOV     #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
15758  061016  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;PRIO:7
15759  061024  104401                          TYPE                    ;REPORT THE POWER FAILURE
15760  061026  061044          $PWRMG: .WORD   $POWER                  ;;POWER FAIL MESSAGE POINTER
15761  061030  012716                          MOV     (PC)+,(SP)      ;;RESTART AT PWRUP
15762  061032  061054          $PWRAD: .WORD   PWRUP                   ;;RESTART ADDRESS
15763  061034  000002                          RTI
15764  061036  000000          $ILLUP: HALT                            ;;THE POWER UP SEQUENCE WAS STARTED
15765  061040  000776                          BR      .-2             ;; BEFORE THE POWER DOWN WAS COMPLETE
15766  061042  000000          $SAVR6: 0                               ;;PUT THE SP HERE
15767  061044  005015  047520  042527  $POWER: .ASCIZ  <15><12>''POWER''
15768  061052  000122
15769                                           .EVEN
15770
15771  061054  012706  001000          PWRUP:  MOV     #STACK,SP       ;RESET SP
15772  061060  005037  177776                  CLR     @#PSW           ;PRIORITY 0 -- CLEAR CODES
15773  061064  000137  001630                  JMP     @#START         ;RESTART PROGRAM
```

```
15774
15775                                   ; ***************************************************************
15776                                           .SBTTL "T" BIT SERVICE ROUTINE
15777                                   ; ***************************************************************
15778
15779  061070  062716  000002          TBSER:  ADD     #2,(SP)         ;MOVE RETURN PC AROUND ERROR CALL
15780  061074  042766  000020  000002          BIC     #20,2(SP)       ;TURN OFF THE "T" BIT
15781  061102  000006                          RTT                     ;RETURN TO THE CALLING TEST
15782
15783                                   .SBTTL  MICROBREAK TRAP SERVICE ROUTINE
15784                                   ;;*************************************************************
15785                                   ;*      THIS ROUTINE MERELY SETS A FLAG
15786                                   ;*      WHEN THE ROUTINE HAS BEEN ENTERED
15787                                   ;*
15788  061104  005237  061112          BKROUT: INC     BKFLAG          ;SET MICROBREAK FLAG TO
15789                                                                   ;INDICATE TRAP TO 4 OCCURRED
15790  061110  000002                          RTI                     ;RETURN FROM TRAP
15791  061112  000000                  BKFLAG: .WORD   0               ;MICROBREAK TRAP FLAG
15792
15793                                   ; ***************************************************************
15794                                           .SBTTL  RSVD INSTRUCTION TRAP SERVICE ROUTINE
15795                                   ; ***************************************************************
15796
15797                                   ;THIS ROUTINE SERVICES UNEXPECTED RESERVED INSTRUCTION TRAP ERRORS
15798                                   ;IT RESULTS IN PRINTING THE ERROR MESSAGE: "TRAPPED TO 10 PC=XXXXXX"
15799                                   ;WHERE XXXXXX IS THE ADDRESS CONTAINING THE INSTRUCTION WORD THAT
15800                                   ;SPRUNG THE TRAP.  AFTER PRINTING THE ERROR MESSAGE AN ATTEMPT IS
15801                                   ;MADE TO RESTART THE PROGRAM AT THE BEGINNING.
15802
15803                                   ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A
15804                                   ;PREVIOUS RSVD INSTRUCTION TRAP OR AN UNEXPECTED BUS ERROR THE PROGRAM
15805                                   ;WILL HALT.  AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE
15806                                   ;TO THE TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15807
15808                                           ;[SP]   PC+2    OF 2ND TRAP
15809                                           ;[SP]+2 PSW       "
15810                                           ;[SP]+4 PC+2    OF 1ST TRAP
15811                                           ;[SP]+6 PSW       "
15812
15813                                           ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE FOLLOWING
15814                                           ;INFORMATION:
15815
15816                                                   ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15817                                                                   ;TRAP (PC AT TIME OF ERROR HALT INDICATES
15818                                                                   ;WHICH OCCURRED FIRST)
15819                                                   ;[CATERR]=2     TWO SUCCESSIVE BUS ERROR TRAPS
15820                                                   ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15821
15822                                   ;THE CONTENTS OF R0 AT THE TIME OF THE
15823                                   ;HALT PROVIDES FURTHER INFORMATION AS TO THE LAST TEST BEING EXECUTED
15824                                   ;WHEN THE TRAPS OCCURRED.
15825
15826                                   ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION
15827                                   ;TESTS TO VERIFY THE RSVD INSTR TRAP MECHANISM PRIOR TO ACTIVATING THE SERVICE
15828                                   ;ROUTINE
15829
```

```
15830  061114  005137  063246          RSVTST: COM     RSVFLG          ;SET RSVD INSTR TRAP TEST FLAG
15831  061120  000002                          RTI                     ;RETURN TO BASIC TEST
15832
15833  061122  005737  063252          RSERR:  TST     @#CATERR        ;ANY PENDING CATASTROPHIC ERRORS
15834  061126  001025                          BNE     INCRSV          ;BE IF YES
15835  061130  105237  063253                  INCB    @#1+CATERR      ;SET RSVD INSTR FLAG
15836  061134  032777  010000  117676          BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT ?
15837  061142  001015                          BNE     RESTAR          ;BR IF YES
15838  061144  104401                          TYPE                    ;GO TYPE "TRAPPED TO 10 PC="
15839  061146  065233                          RSMSG
15840  061150  011646                  RSBERT: MOV     (SP),-(SP)      ;GET ERROR PC ON STACK FOR PRINTING
15841  061152  104402                          TYPOC                   ;TYPE THE ERROR PC
15842  061154  104401                          TYPE                    ;OUTPUT CR / LF
15843  061156  001115                          $CRLF
15844  061160  005237  001012                  INC     @#$ERTTL        ;COUNT THE ERROR
15845  061164  032777  100000  117646          BIT     #BIT15,@SWR     ;HALT ON ERROR?
15846  061172  001401                          BEQ     RESTAR          ;BR IF NOT
15847  061174  000000                          HALT                    ;HALT ON ERROR--PRESS CONTINUE TO RESTART
15848  061176  000137  003262          RESTAR: JMP     @#INIT          ;GO ATTEMPT RESTART
15849  061202  105237  063253          INCRSV: INCB    @#1+CATERR      ;INCREMENT RSVD INSTR FLAG
15850  061206  000000                          HALT                    ;CATASTROPHIC ERROR HALT
15851  061210  000772                          BR      RESTAR          ;DEPRESSING CONTINUE WILL CAUSE
15852                                                                  ;ATTEMPT TO RESTART.
15853
15854                                  ; ******************************************************************
15855                                          .SBTTL  BUS ERROR TRAP SERVICE ROUTINE
15856                                  ; ******************************************************************
15857
15858                                  ;THIS ROUTINE SERVICES UNEXPECTED BUS ERROR TRAPS (BUS TIMEOUT, ODD ADDRESS
15859                                  ;ERRORS, STACK OVERFLOW, AND ILLEGAL INSTRUCTIONS). IT RESULTS IN PRINTING THE
15860                                  ;ERROR MESSAGE: "TRAPPED TO 4 PC =XXXXXX" WHERE XXXXXX IS THE
15861                                  ;CONTENTS OF THE PC WHEN THE TRAP WAS SPRUNG.  AFTER PRINTING THE
15862                                  ;ERROR MESSAGE AN ATTEMPT IS MADE TO RESTART THE PROGRAM AT
15863                                  ;THE BEGINNING.
15864
15865                                  ;IF THE TRAP IS SPRUNG WHILE IN THE PROCESS OF TRYING TO SERVICE A PREVIOUS
15866                                  ;RSVD INSTR TRAP OR A PREVIOUS BUS ERROR, THE PROGRAM WILL HALT.
15867                                  ;AFTER THE HALT THE STACK WILL CONTAIN INFORMATION RELATIVE TO THE
15868                                  ;TWO SUCCESSIVE TRAPS AS SHOWN BELOW:
15869
15870                                          ;[SP]   PC+2    OF 2ND TRAP
15871                                          ;[SP]+2 PSW             "
15872                                          ;[SP]+4 PC+2    OF 1ST TRAP
15873                                          ;[SP]+6 PSW             "
15874
15875                                          ;LOCATION "CATERR" CAN BE EXAMINED TO OBTAIN THE LOLLOWING
15876                                          ;INFORMATION:
15877
15878                                                  ;[CATERR]=401   RSVD INSTR TRAP COMBINED WITH A BUS ERROR
15879                                                                  ;TRAP (PC AT TIME OF ERROR HALT
15880                                                                  ;INDICATES WHICH OCCURRED FIRST)
15881                                                  ;[CATERR]=2     TWO SUCCESSIVE BUS ERRORS
15882                                                  ;[CATERR]=1000  TWO SUCCESSIVE RSVD INSTR TRAPS
15883
15884                                  ;THE CONTENTS OF R0 AT THE TIME OF
15885                                  ;THE HALT PROVIDED FURTHER INFORMATION AS TO THE TEST IN PROGRESS
```

```
                                        ;WHEN THE TRAPS OCCURRED.
                                        ;
                                        ;THE CONTENTS OF THE SP CAN BE USED TO INDICATE IF STACK OVERFLOW CAUSED
                                        ;THE BUSS ERROR TRAP(S) AS SHOWN BELOW:
                                        ;
                                                ;400>[SP]>336   YELLOW ZONE
                                                ;[SP]=0         RED ZONE
                                        ;
                                        ;THESE TWO INSTRUCTIONS ARE USED BY THE BASIC INSTRUCTION TESTS TO
                                        ;VERIFY THAT THE BUS ERROR TRAP MECHANISM WORKS PRIOR TO ACTIVATING
                                        ;THE SERVICE ROUTINE
                                        ;
061212  005137  063250        BETST:  COM     BERFLG          ;SET BUS ERROR TRAP TEST FLAG
061216  000002                        RTI                     ;RETURN TO BASIC TEST
                                        ;
061220  005737  063252        BERR:   TST     @#CATERR        ;ANY CATASTROPHIC ERRORS PENDING?
061224  001011                        BNE     2$              ;BR IF YES
061226  105237  063252                INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
061232  032777  010000  117600        BIT     #SW12,@SWR      ;INHIBIT ERROR PRINT
061240  001356                        BNE     RESTAR          ;BR IF YES
061242  104401                        TYPE                    ;PRINT "TRAP TO 4" MESSAGE
061244  065206                        BEMSG
061246  000740                        BR      RSBERT          ;TYPE REST OF BUS ERROR MESSAGE
                                        ;
061250  105237  063252        2$:     INCB    @#CATERR        ;SET CATASTROPHIC ERROR FLAG
061254  000000                        HALT                    ;CATASTROPHIC ERROR HALT-SCHOOLS OUT
061256  000747                        BR      RESTAR          ;DEPRESS CONTINUE TO ATTEMPT RESTART
                                        ;
                                        ;
                                        ;
                                .SBTTL  SCOPE HANDLER ROUTINE
                                        ;
                                        ;;*********************************************************************
                                        ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
                                        ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
                                        ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
                                        ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
                                        ;*SW14=1         LOOP ON TEST
                                        ;*SW11=1         INHIBIT ITERATIONS
                                        ;*SW09=1         LOOP ON ERROR
                                        ;*CALL
                                        ;*      SCOPE           ;;SCOPE=IOT
                                        ;
061260                          $SCOPE:
061260  020037  001124                  CMP     R0,@#$TESTN     ;ANY MISSED TESTS ?
061264  001406                        BEQ     10$             ;BR IF NOT
061266  012737  061276  001112        MOV     #12$,@#$ESCAPE  ;NO ERROR LOOPING
061274  104011                        ERROR   11              ;MISSED TESTS ERROR CALL
061276  005037  001112        12$:    CLR     @#$ESCAPE       ;NORMAL ERROR LOOPING
061302  110037  001002        10$:    MOVB    R0,@#$TSTNM     ;INSURE TSTNUM IS CORRECT
061306  032777  002000  117524        BIT     #SW10,@SWR      ;LOOP ON SELECTED TEST?
061314  001411                        BEQ     11$             ;BR IF NO
061316  017737  117516  063242        MOV     @SWR,@#SELTST   ;GET CONTENTS OF SWITCHES
061324  042737  177000  063242        BIC     #177000,@#SELTST        ;MASK OUT SWR<15:9>
061332  020037  063242                CMP     R0,@#SELTST     ;IS THIS THE SELECTED TEST?
061336  001510                        BEQ     $OVER           ;BR IF YES
```

```
15942  061340                                  11$:
15943  061340  032777  040000  117472  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
15944  061346  001104                          BNE     $OVER           ;;YES IF SW14=1
15945                                  ;#####START OF CODE FOR THE XOR TESTER#####
15946  061350  000416                  $XTSTR: BR      6$              ;;IF RUNNING ON THE "XOR" TESTER CHANGE
15947                                                                  ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
15948  061352  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE THE CONTENTS OF THE ERROR VECTOR
15949  061356  012737  061376  000004          MOV     #5$,@#ERRVEC    ;;SET FOR TIMEOUT
15950  061364  005737  177060                  TST     @#177060        ;;TIME OUT ON XOR?
15951  061370  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15952  061374  000453                          BR      $SVLAD          ;;GO TO THE NEXT TEST
15953  061376  022626                  5$:     CMP     (SP)+,(SP)+     ;;CLEAR THE STACK AFTER A TIME OUT
15954  061400  012637  000004                  MOV     (SP)+,@#ERRVEC  ;;RESTORE THE ERROR VECTOR
15955  061404  000413                          BR      7$              ;;LOOP ON THE PRESENT TEST
15956  061406                          6$:;#####END OF CODE FOR THE XOR TESTER#####
15957  061406  105737  001003          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
15958  061412  001421                          BEQ     3$              ;;BR IF NO
15959  061414  123737  001015  001003          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
15960  061422  101015                          BHI     3$              ;;BR IF NO
15961  061424  032777  001000  117406          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
15962  061432  001404                          BEQ     4$              ;;BR IF NO
15963  061434  013737  001010  001006  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
15964  061442  000446                          BR      $OVER
15965  061444  105037  001003          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
15966  061450  005037  001110                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
15967  061454  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
15968  061456  032777  004000  117354  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
15969  061464  001011                          BNE     1$              ;;BR IF YES
15970  061466  005737  001126                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
15971  061472  001406                          BEQ     1$              ;;      INHIBIT ITERATIONS
15972  061474  005237  001004                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
15973  061500  023737  001110  001004          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
15974  061506  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
15975  061510  012737  000001  001004  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
15976  061516  013737  061610  001110          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
15977  061524  105237  001002          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
15978  061530  113737  001002  001124          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
15979  061536  011637  001006                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
15980  061542  011637  001010                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
15981  061546  005037  001112                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
15982  061552  112737  000001  001015          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
15983  061560  013777  001002  117254  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
15984  061566  013716  001006                  MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
15985  061572  120037  001002                  CMPB    R0,@#$TSTNM     ;WAS $TSTNM INCREMENTED?
15986  061576  001401                          BEQ     10$             ;BR IF NOT
15987  061600  005200                          INC     R0              ;INCREMENT TEST NUMBER
15988  061602  010037  001124          10$:    MOV     R0,@#$TESTN     ;FIX $TESTN TO BE WORD COUNT, NOT BYTE
15989  061606  000002                          RTI
15990  061610  000200                  $MXCNT: 200                     ;;MAX. NUMBER OF ITERATIONS
15991
15992  061612  005137  063244          SCOPEA: COM     @#SCOFLG        ;THESE TWO INSTRUCTIONS ARE
15993  061616  000002                          RTI                     ;USED IN THE BASIC TESTS TO
15994                                                                  ;VERIFY THE IOT LINKAGE
15995
15996
15997
```

```
                                .SBTTL  ERROR HANDLER ROUTINE

                                ;;*****************************************************************
                                ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
                                ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
                                ;*AND GO TO $ERRTYP ON ERROR
                                ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
                                ;*SW15=1         HALT ON ERROR
                                ;*SW13=1         INHIBIT ERROR TYPEOUTS
                                ;*SW09=1         LOOP ON ERROR
                                ;*CALL
                                ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER

061620                          $ERROR:
061620  010546                         MOV     R5,-(SP)        ;SAVE R5 ON STACK
061622  012705  001060                 MOV     #$REGAD,R5      ;GET POINTER
061626  016625  000004                 MOV     4(SP),(R5)+     ;SAVE ERROR PSW IN $REGAD FOR TYPOUT
061632  010025                         MOV     R0,(R5)+        ;SAVE R0 FOR TYPEOUTS
061634  010125                         MOV     R1,(R5)+        ;SAVE R1 IN $REG1
061636  010225                         MOV     R2,(R5)+        ;SAVE R2 IN $REG2, ETC.
061640  010325                         MOV     R3,(R5)+
061642  010425                         MOV     R4,(R5)+
061644  022715  177777                 CMP     #-1,(R5)        ;IS SP  ALREADY STORED IN $REG5?
061650  001001                         BNE     10$             ;BR IF YES
061652  010615                         MOV     SP,(R5)         ;PUT SP IN $REG5 FOR TYPOUT
061654  012605                 10$:    MOV     (SP)+,R5        ;RESTORE R5
061656  105237  001003         7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
061662  001775                         BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
061664  013777  001002  117150         MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
061672  005237  001012                 INC     $ERTTL          ;;INC THE ERROR COUNT
061676  011637  001016                 MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
061702  162737  000002  001016         SUB     #2,$ERRPC
061710  117737  117102  001014         MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
061716  032777  020000  117114         BIT     #BIT13,@SWR     ;;SKIP TYPEOUT IF SET
061724  001004                         BNE     20$             ;;SKIP TYPEOUTS
061726  004737  062046                 JSR     PC,$ERRTYP      ;;GO TO USER ERROR ROUTINE
061732  104401  001115                 TYPE    ,$CRLF
061736                          20$:
061736  122737  000001  001140         CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
061744  001007                         BNE     2$              ;;NO,SKIP APT ERROR REPORT
061746  113737  001014  061760         MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
061754  004737  062736                 JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
061760     000                  21$:    .BYTE   0
061761     000                          .BYTE   0
061762  000777                 22$:    BR      22$             ;;APT ERROR LOOP
061764  005777  117050         2$:     TST     @SWR            ;;HALT ON ERROR
061770  100001                         BPL     3$              ;;SKIP IF CONTINUE
061772  000000                         HALT                    ;;HALT ON ERROR!
061774  032777  001000  117036 3$:     BIT     #BIT09,@SWR     ;;LOOP ON ERROR SWITCH SET?
062002  001402                         BEQ     4$              ;;BR IF NO
062004  013716  001010                 MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
062010  005737  001112         4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
062014  001402                         BEQ     5$              ;;BR IF NONE
062016  013716  001112                 MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
062022                          5$:
062022  012737  177777  001074         MOV     #-1,@#$REG5     ;FLAG CURRENT STACK POINTER TO BE TYPED
```

```
16054  062030  042766  000020  000002          BIC     #20,2(SP)       ;CLEAR T BIT IN CASE ERROR OCCURED
16055                                                                   ;IN T BIT TESTS
16056  062036  000002                          RTI
16057
16058  062040  005137  063240          ERRA:   COM     @#ERRFLG        ;THESE TWO INSTRUCTIONS ARE USED
16059  062044  000002                          RTI                     ;IN THE BASIC TESTS TO VERIFY THE EMT
16060
16061                                  .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE
16062
16063                                  ;;**********************************************************************
16064                                  ;*THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
16065                                  ;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE "ERROR TABLE" ($ERRTB),
16066                                  ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
16067
16068  062046                          $ERRTYP:
16069  062046  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16070  062052  010046                          MOV     R0,-(SP)        ;;SAVE R0
16071  062054  005000                          CLR     R0              ;;PICKUP THE ITEM INDEX
16072  062056  153700  001014                  BISB    @#$ITEMB,R0
16073  062062  001004                          BNE     1$              ;;IF ITEM NUMBER IS ZERO, JUST
16074                                                                  ;;TYPE THE PC OF THE ERROR
16075  062064  013746  001016                  MOV     $ERRPC,-(SP)    ;;SAVE $ERRPC FOR TYPEOUT
16076                                                                  ;;ERROR ADDRESS
16077  062070  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16078  062072  000426                          BR      6$              ;;GET OUT
16079  062074  005300                  1$:     DEC     R0              ;;ADJUST THE INDEX SO THAT IT WILL
16080  062076  006300                          ASL     R0              ;;      WORK FOR THE ERROR TABLE
16081  062100  006300                          ASL     R0
16082  062102  006300                          ASL     R0
16083  062104  062700  001150                  ADD     #$ERRTB,R0      ;;FORM TABLE POINTER
16084  062110  012037  062120                  MOV     (R0)+,2$        ;;PICKUP "ERROR MESSAGE" POINTER
16085  062114  001404                          BEQ     3$              ;;SKIP TYPEOUT IF NO POINTER
16086  062116  104401                          TYPE                    ;;TYPE THE "ERROR MESSAGE"
16087  062120  000000                  2$:     .WORD   0               ;;"ERROR MESSAGE" POINTER GOES HERE
16088  062122  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16089  062126  012037  062136          3$:     MOV     (R0)+,4$        ;;PICKUP "DATA HEADER" POINTER
16090  062132  001404                          BEQ     5$              ;;SKIP TYPEOUT IF 0
16091  062134  104401                          TYPE                    ;;TYPE THE "DATA HEADER"
16092  062136  000000                  4$:     .WORD   0               ;;"DATA HEADER" POINTER GOES HERE
16093  062140  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16094  062144  011000                  5$:     MOV     (R0),R0         ;;PICKUP "DATA TABLE" POINTER
16095  062146  001004                          BNE     7$              ;;GO TYPE THE DATA
16096  062150  012600                  6$:     MOV     (SP)+,R0        ;;RESTORE R0
16097  062152  104401  001115                  TYPE    ,$CRLF          ;;"CARRIAGE RETURN" & "LINE FEED"
16098  062156  000207                          RTS     PC              ;;RETURN
16099  062160                          7$:
16100  062160  013046                          MOV     @(R0)+,-(SP)    ;;SAVE @(R0)+ FOR TYPEOUT
16101  062162  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
16102  062164  005710                          TST     (R0)            ;;IS THERE ANOTHER NUMBER?
16103  062166  001770                          BEQ     6$              ;;BR IF NO
16104  062170  104401  062176                  TYPE    ,8$             ;;TYPE TWO(2) SPACES
16105  062174  000771                          BR      7$              ;;LOOP
16106  062176  020040      000         8$:     .ASCIZ  /  /            ;;TWO(2) SPACES
16107          062202                          .EVEN
16108
16109                                  ; **********************************************************************
```

```
16110                                   .SBTTL PRINT ROUTINES
16111                                   ; ******************************************************************
16112
16113  062202  005137  063236  PRINA:  COM     @#PRIFLG        ;THESE TWO INSTRUCTIONS ARE
16114  062206  000002                  RTI                     ;USED BY THE BASIC TESTS TO VERIFY
16115                                                           ;THE TRAP INSTRUCTION
16116
16117                                   .SBTTL  TYPE ROUTINE
16118
16119                                   ;;******************************************************************
16120                                   ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
16121                                   ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
16122                                   ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
16123                                   ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
16124                                   ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
16125                                   ;*
16126                                   ;*CALL:
16127                                   ;*1) USING A TRAP INSTRUCTION
16128                                   ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
16129                                   ;*OR
16130                                   ;*      TYPE
16131                                   ;*      MESADR
16132                                   ;*
16133
16134  062210  105737  001057  $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
16135  062214  100002                  BPL     1$              ;;BR IF YES
16136  062216  000000                  HALT                    ;;HALT HERE IF NO TERMINAL
16137  062220  000430                  BR      3$              ;;LEAVE
16138  062222  010046          1$:     MOV     R0,-(SP)        ;;SAVE R0
16139  062224  017600  000002          MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
16140  062230  122737  000001  001140  CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
16141  062236  001011                  BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
16142  062240  132737  000100  001141  BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
16143  062246  001405                  BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
16144  062250  010037  062260          MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
16145  062254  004737  062726          JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
16146  062260  000000          61$:    .WORD   0               ;;MESSAGE ADDRESS
16147  062262  132737  000040  001141  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED
16148  062270  001003                  BNE     60$             ;;YES,SKIP TYPE OUT
16149  062272  112046          2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
16150  062274  001005                  BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
16151  062276  005726                  TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
16152  062300  012600          60$:    MOV     (SP)+,R0        ;;RESTORE R0
16153  062302  062716  000002  3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
16154  062306  000002                  RTI                     ;;RETURN
16155  062310  122716  000011  4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
16156  062314  001430                  BEQ     8$
16157  062316  122716  000200          CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
16158  062322  001006                  BNE     5$
16159  062324  005726                  TST     (SP)+           ;;POP  <CR><LF> EQUIV
16160  062326  104401                  TYPE                    ;;TYPE A CR AND LF
16161  062330  001115                  $CRLF
16162  062332  105037  062466          CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
16163  062336  000755                  BR      2$              ;;GET NEXT CHARACTER
16164  062340  004737  062422  5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
16165  062344  123726  001056  6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
```

```
16166  062350  001350                        BNE     2$              ;;IF NO GO GET NEXT CHAR.
16167  062352  013746  001054                MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
16168                                                                 ;;AND THE NULL CHAR.
16169  062356  105366  000001        7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
16170  062362  002770                        BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
16171  062364  004737  062422                JSR     PC,$TYPEC       ;;GO TYPE A NULL
16172  062370  105337  062466                DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
16173  062374  000770                        BR      7$              ;;LOOP
16174
16175                                        ;HORIZONTAL TAB PROCESSOR
16176
16177  062376  112716  000040        8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
16178  062402  004737  062422        9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
16179  062406  132737  000007  062466        BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
16180  062414  001372                        BNE     9$              ;;TAB STOP
16181  062416  005726                        TST     (SP)+           ;;POP SPACE OFF STACK
16182  062420  000724                        BR      2$              ;;GET NEXT CHARACTER
16183  062422  105777  116422        $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
16184  062426  100375                        BPL     $TYPEC
16185  062430  116677  000002  116414        MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
16186  062436  122766  000015  000002        CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
16187  062444  001003                        BNE     1$              ;;BRANCH IF NO
16188  062446  105037  062466                CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
16189  062452  000406                        BR      $TYPEX          ;;EXIT
16190  062454  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
16191  062462  001402                        BEQ     $TYPEX          ;;BRANCH IF YES
16192  062464  105227                        INCB    (PC)+           ;;COUNT THE CHARACTER
16193  062466  000000        $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
16194  062470  000207        $TYPEX: RTS     PC
16195
16196
16197                                        .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
16198
16199                                        ;;**********************************************************************
16200                                        ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
16201                                        ;*OCTAL (ASCII) NUMBER AND TYPE IT.
16202                                        ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
16203                                        ;*CALL:
16204                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16205                                        ;*      TYPOS                   ;;CALL FOR TYPEOUT
16206                                        ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
16207                                        ;*      .BYTE   M               ;;M=1 OR 0
16208                                        ;*                                      ;;1=TYPE LEADING ZEROS
16209                                        ;*                                      ;;0=SUPPRESS LEADING ZEROS
16210                                        ;*
16211                                        ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
16212                                        ;*$TYPOS OR $TYPOC
16213                                        ;*CALL:
16214                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16215                                        ;*      TYPON                   ;;CALL FOR TYPEOUT
16216                                        ;*
16217                                        ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
16218                                        ;*CALL:
16219                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
16220                                        ;*      TYPOC                   ;;CALL FOR TYPEOUT
16221
```

```
16222  062472  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
16223  062476  116637  000001  062715          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
16224  062504  112637  062717                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
16225  062510  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
16226  062514  000406                          BR      $TYPON
16227  062516  112737  000001  062715  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
16228  062524  112737  000006  062717          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
16229  062532  112737  000005  062714  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
16230  062540  010346                          MOV     R3,-(SP)        ;;SAVE R3
16231  062542  010446                          MOV     R4,-(SP)        ;;SAVE R4
16232  062544  010546                          MOV     R5,-(SP)        ;;SAVE R5
16233  062546  113704  062717                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
16234  062552  005404                          NEG     R4
16235  062554  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
16236  062560  110437  062716                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
16237  062564  113704  062715                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
16238  062570  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
16239  062574  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
16240  062576  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
16241  062600  000404                          BR      3$              ;;GO DO MSB
16242  062602  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
16243  062604  006105                          ROL     R5
16244  062606  006105                          ROL     R5
16245  062610  010503                          MOV     R5,R3
16246  062612  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
16247  062614  105337  062716                  DECB    $OMODE          ;;TYPE THIS DIGIT?
16248  062620  100016                          BPL     7$              ;;BR IF NO
16249  062622  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
16250  062626  001002                          BNE     4$              ;;TEST FOR 0
16251  062630  005704                          TST     R4              ;;SUPPRESS THIS 0?
16252  062632  001403                          BEQ     5$              ;;BR IF YES
16253  062634  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
16254  062636  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
16255  062642  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
16256  062646  110337  062712                  MOVB    R3,8$           ;;SAVE FOR TYPING
16257  062652  104401  062712                  TYPE    ,8$             ;;GO TYPE THIS DIGIT
16258  062656  105337  062714          7$:     DECB    $OCNT           ;;COUNT BY 1
16259  062662  003347                          BGT     2$              ;;BR IF MORE TO DO
16260  062664  002402                          BLT     6$              ;;BR IF DONE
16261  062666  005204                          INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
16262  062670  000744                          BR      2$              ;;GO DO THE LAST DIGIT
16263  062672  012605                  6$:     MOV     (SP)+,R5        ;;RESTORE R5
16264  062674  012604                          MOV     (SP)+,R4        ;;RESTORE R4
16265  062676  012603                          MOV     (SP)+,R3        ;;RESTORE R3
16266  062700  016666  000002  000004          MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
16267  062706  012616                          MOV     (SP)+,(SP)
16268  062710  000002                          RTI                     ;;RETURN
16269  062712     000          8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
16270  062713     000                  .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
16271  062714     000          $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
16272  062715     000          $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
16273  062716  000000          $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
16274
16275                          .SBTTL  APT COMMUNICATIONS ROUTINE
16276
16277                          ;;**************************************************************
```

```
16278  062720  112737  000001  063164  $ATY1:  MOVB    #1,$FFLG        ;;TO REPORT FATAL ERROR
16279  062726  112737  000001  063162  $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
16280  062734  000403                  $ATY4:  BR      $ATYC
16281  062736  112737  000001  063164          MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
16282  062744                          $ATYC:
16283  062744  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
16284  062746  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
16285  062750  105737  063162                  TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
16286  062754  001450                          BEQ     5$              ;;IF NOT:  BR
16287  062756  122737  000001  001140          CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
16288  062764  001031                          BNE     3$              ;;IF NOT:  BR
16289  062766  132737  000100  001141          BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
16290  062774  001425                          BEQ     3$              ;;IF NOT:  BR
16291  062776  017600  000004                  MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
16292  063002  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16293  063010  005737  001120          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
16294  063014  001375                          BNE     1$              ;;IF NOT:  WAIT
16295  063016  010037  001134                  MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
16296  063022  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
16297  063024  001376                          BNE     2$
16298  063026  163700  001134                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
16299  063032  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
16300  063034  010037  001136                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
16301  063040  012737  000004  001120          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
16302  063046  000413                          BR      5$
16303  063050  017637  000004  063074  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
16304  063056  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
16305  063064  013746  177776                  MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
16306  063070  004737  062210                  JSR     PC,$TYPE        ;;CALL TYPE MACRO
16307  063074  000000                  4$:     .WORD   0
16308  063076                          5$:
16309  063076  105737  063164          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
16310  063102  001416                          BEQ     12$             ;;IF NOT:  BR
16311  063104  005737  001140                  TST     $ENV            ;;RUNNING UNDER APT?
16312  063110  001413                          BEQ     12$             ;;IF NOT:  BR
16313  063112  005737  001120          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
16314  063116  001375                          BNE     11$             ;;IF NOT:  WAIT
16315  063120  017637  000004  001122          MOV     @4(SP),$FATAL   ;;GET ERROR #
16316  063126  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
16317  063134  005237  001120                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
16318  063140  105037  063164          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
16319  063144  105037  063163                  CLRB    $LFLG           ;;CLEAR LOG FLAG
16320  063150  105037  063162                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
16321  063154  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
16322  063156  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
16323  063160  000207                          RTS     PC              ;;RETURN
16324  063162     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
16325  063163     000          $LFLG:  .BYTE   0               ;;LOG FLAG
16326  063164     000          $FFLG:  .BYTE   0               ;;FATAL FLAG
16327          063166                  .EVEN
16328          000200          APTSIZE=200
16329          000001          APTENV=001
16330          000100          APTSPOOL=100
16331          000040          APTCSUP=040
16332
16333                          .SBTTL  TRAP DECODER
```

```
16334
16335                                   ;;*****************************************************************
16336                                   ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
16337                                   ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
16338                                   ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
16339                                   ;*GO TO THAT ROUTINE.
16340
16341  063166  010046                  $TRAP:  MOV     R0,-(SP)          ;;SAVE R0
16342  063170  016600  000002                  MOV     2(SP),R0          ;;GET TRAP ADDRESS
16343  063174  005740                          TST     -(R0)             ;;BACKUP BY 2
16344  063176  111000                          MOVB    (R0),R0           ;;GET RIGHT BYTE OF TRAP
16345  063200  006300                          ASL     R0                ;;POSITION FOR INDEXING
16346  063202  016000  063222                  MOV     $TRPAD(R0),R0     ;;INDEX TO TABLE
16347  063206  000200                          RTS     R0                ;;GO TO ROUTINE
16348
16349
16350                                   ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
16351
16352  063210  011646                  $TRAP2: MOV     (SP),-(SP)        ;;MOVE THE PC DOWN
16353  063212  016666  000004  000002          MOV     4(SP),2(SP)       ;;MOVE THE PSW DOWN
16354  063220  000002                          RTI                       ;;RESTORE THE PSW
16355
16356                                   .SBTTL  TRAP TABLE
16357
16358                                   ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
16359                                   ;*BY THE "TRAP" INSTRUCTION.
16360
16361                                   ;       ROUTINE
16362                                   ;       -------
16363  063222  063210                  $TRPAD: .WORD   $TRAP2
16364  063224  062210                          $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROU INE
16365  063226  062516                          $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
16366  063230  062472                          $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
16367  063232  062532                          $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
16368
16369
16370
16371                                   ;FLAGS, CONSTANTS, AND VARIABLES
16372
16373  063234  000000                  BPTLOC: 0                         ;STORES 16 USER DEFINED MAINTENANCE
16374                                                                    ;BREAKPOINTS
16375  063236  000000                  PRIFLG: 0                         ;FLAG USED BY BASIC TESTS FOR TRAP TEST
16376  063240  000000                  ERRFLG: 0                         ;FLAG USED BY BASIC TESTS FOR EMT TEST
16377  063242  000000                  SELTST: 0                         ;STORES SR<8:0> FOR LOOP ON SELECTED TEST
16378  063244  000000                  SCOFLG: 0                         ;USED BY BASIC TESTS FOR IOT TEST
16379  063246  000000                  RSVFLG: 0                         ;FLAG USED BY BASIC TEST OF RSVD INSTR TRAP
16380  063250  000000                  BERFLG: 0                         ;FLAG USED BY BASIC TEST OF BUS ERROR TRAPS
16381  063252  000000                  CATERR: 0                         ;FLAGS USED BY BUS ERROR AND RSVD INSTR TRAP
16382                                                                    ;SERVICE ROUTINES
16383  063254  000000                  ONCE:   0                         ;FLAGS PROGRAM TITLE HAS BEEN PRINTED
16384                                   ;COMMON DATA STRUCTURES AND MISCELLANEOUS TABLES
16385
16386  063256  177400                  OBUF:   177400                    ;DL11 OUTPUT TEST BUFFER
16387  063260  177400                          177400
16388  063262  177400                          177400
16389  063264  177400                          177400
```

```
16390
16391  063266  000004                  IBUF:   .BLKW   4               ;DL11 INPUT TEST BUFFER
16392
16393  063276  063322                  ATA:    DWTA
16394  063300  064032                          DWTB
16395  063302  064630                          DBTA
16396  063304  064634                          DBTB
16397  063306  063312                          MBUF0
16398  063310  063316                          MBUF1
16399
16400  063312  000000                  MBUF0:  0
16401  063314  000000                          0
16402  063316  000000                  MBUF1:  0
16403  063320  000000                          0
16404  063322  000000                  DWTA:   0
16405  063324  177777                          -1
16406  063326  177400                          177400
16407  063330  000377                          377
16408  063332  125252                          125252
16409  063334  052525                  ALUADD: 052525                  ;ALSO SERVES AS NULL ENTRY FOR ALUADD
16410
16411                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU ADD TEST IN THE
16412                                  ;COMBINED INSTRUCTION TESTS
16413
16414  063336  000000                          000000                  ;SRC OP1
16415  063340  000000                          000000                  ;DST OP1
16416  063342  000000                          000000                  ;ANS1
16417  063344  177777                          177777                  ;SRC OP2
16418  063346  177777                          177777                  ;DST OP2
16419  063350  177776                          177776                  ;ANS2
16420  063352  125252                          125252                  ;SRC OP3
16421  063354  052525                          052525                  ;DST OP3
16422  063356  177777                          177777                  ;ANS3
16423  063360  052525                          052525                  ;SRC OP4
16424  063362  125252                          125252                  ;DST OP4
16425  063364  177777                          177777                  ;ANS4
16426  063366  125252                          125252                  ;SRC OP5
16427  063370  125252                          125252                  ;DST OP5
16428  063372  052524                          052524                  ;ANS5
16429  063374  052525                          052525                  ;SRC OP6
16430  063376  052525                          052525                  ;DST OP6
16431  063400  125252                          125252                  ;ANS6
16432  063402  052525                          052525                  ;SRC OP7
16433  063404  125253                          125253                  ;DST OP7
16434  063406  000000                          000000                  ;ANS7
16435  063410  125253                          125253                  ;SRC OP8
16436  063412  052525                          052525                  ;DST OP8
16437  063414  000000                  ANDTAB: 000000                  ;ANS8 -- ALSO NULL ENTRY FOR ANDTAB
16438
16439                                  ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "AND" TESTS IN THE
16440                                  ;COMBINED INSTRUCTION EXERCISER TESTS
16441
16442  063416  000000                          000000                  ;SRC OP1
16443  063420  000000                          000000                  ;DEST OP1
16444  063422  000000                          000000                  ;ANS1
16445  063424  177777                          177777                  ;SRC OP2
```

```
16446  063426  177777          177777          ;DST OP2
16447  063430  000000          000000          ;ANS2
16448  063432  000000          000000          ;SRC OP3
16449  063434  177777          177777          ;DST OP3
16450  063436  177777          177777          ;ANS3
16451  063440  177777          177777          ;SRC OP4
16452  063442  000000          000000          ;DST OP4
16453  063444  000000          000000          ;ANS4
16454  063446  125252          125252          ;SRC OP5
16455  063450  125252          125252          ;DST OP5
```

```
16456  063452  000000                    000000                  ;ANS5
16457  063454  052525                    052525                  ;SRC OP6
16458  063456  052525                    052525                  ;DST OP6
16459  063460  000000                    000000                  ;ANS6
16460  063462  125252                    125252                  ;SRC OP7
16461  063464  052525                    052525                  ;DST OP7
16462  063466  052525                    052525                  ;ANS7
16463  063470  052525                    052525                  ;SRC OP8
16464  063472  125252                    125252                  ;DST OP8
16465  063474  125252            ORTAB:  125252                  ;ANS8 -- ALSO NULL ENTRY FOR ORTAB
16466
16467                            ;THIS TABLE OF 8 ENTRIES IS USED BY THE ALU "OR" TEST IN THE
16468                            ;COMBINED INSTRUCTION EXERCISER TEST
16469
16470  063476  000000                    000000                  ;SRC OP1
16471  063500  000000                    000000                  ;DEST OP1
16472  063502  000000                    000000                  ;ANS1
16473  063504  177777                    177777                  ;SRC OP2
16474  063506  177777                    177777                  ;DST OP2
16475  063510  177777                    177777                  ;ANS2
16476  063512  000000                    000000                  ;SRC OP3
16477  063514  177777                    177777                  ;DST OP3
16478  063516  177777                    177777                  ;ANS3
16479  063520  177777                    177777                  ;SRC OP4
16480  063522  000000                    000000                  ;DST OP4
16481  063524  177777                    177777                  ;ANS4
16482  063526  125252                    125252                  ;SRC OP5
16483  063530  125252                    125252                  ;DST OP5
16484  063532  125252                    125252                  ;ANS5
16485  063534  052525                    052525                  ;SRC OP6
16486  063536  052525                    052525                  ;DST OP6
16487  063540  052525                    052525                  ;ANS6
16488  063542  125252                    125252                  ;SRC OP7
16489  063544  052525                    052525                  ;DST OP7
16490  063546  177777                    177777                  ;ANS7
16491  063550  052525                    052525                  ;SRC OP8
16492  063552  125252                    125252                  ;DST OP8
16493  063554  177777            ALUSUB: 177777                  ;ANS8 -- ALSO NULL ENTRY FOR ALUSUB
16494
16495                            ;THIS TABLE OF 8 ENTRIES IS USED BYTHE ALU SUB TEST IN THE
16496                            ;COMBINED INSTRUCTION EXERCISER TESTS
16497
16498  063556  000000                    000000                  ;SRC OP1
16499  063560  000000                    000000                  ;DST OP1
16500  063562  000000                    000000                  ;ANS1
16501  063564  177777                    177777                  ;SRC OP2
16502  063566  177777                    177777                  ;DST OP2
16503  063570  000000                    000000                  ;ANS2
16504  063572  125252                    125252                  ;SRC OP3
16505  063574  052525                    052525                  ;DST OP3
16506  063576  125253                    125253                  ;ANS3
16507  063600  052525                    052525                  ;SRC OP4
16508  063602  125252                    125252                  ;DST OP4
16509  063604  052525                    052525                  ;ANS4
16510  063606  125252                    125252                  ;SRC OP5
16511  063610  125252                    125252                  ;DST OP5
```

```
16512  063612  000000                  000000                  ;ANS5
16513  063614  052525                  052525                  ;SRC OP6
16514  063616  052525                  052525                  ;DST OP6
16515  063620  000000                  000000                  ;ANS6
16516  063622  052525                  052525                  ;SRC OP7
16517  063624  125253                  125253                  ;DST OP7
16518  063626  052526                  052526                  ;ANS7
16519  063630  125253                  125253                  ;SRC OP8
16520  063632  052525                  052525                  ;DST OP8
16521  063634  125252                  125252                  ;ANS8
16522
16523  063636  005702          INSTAB: TST     R2      ;BEGINNING OF INSTRUCTION TABLE OF INSTRUCTIONS
16524  063640  005002                  CLR     R2      ;THAT  TEST BUT SERVICE IN VARIOUS ROM LOCATIONS
16525  063642  005102                  COM     R2
16526  063644  005202                  INC     R2
16527  063646  005302                  DEC     R2
16528  063650  005502                  ADC     R2
16529  063652  005602                  SBC     R2
16530  063654  006202                  ASR     R2
16531  063656  006302                  ASL     R2
16532  063660  105002                  CLRB    R2
16533  063662  105102                  COMB    R2
16534  063664  105202                  INCB    R2
16535  063666  105302                  DECB    R2
16536  063670  105502                  ADCB    R2
16537  063672  105502                  ADCB    R2
16538  063674  105602                  SBCB    R2
16539  063676  105702                  TSTB    R2
16540  063700  106202                  ASRB    R2
16541  063702  106302                  ASLB    R2
16542  063704  151302                  BISB    (R3),R2
16543  063706  074302                  XOR     R3,R2
16544  063710  121302                  CMPB    (R3),R2
16545  063712  131302                  BITB    (R3),R2
16546  063714  141302                  BICB    (R3),R2
16547  063716  111302                  MOVB    (R3),R2
16548  063720  021302                  CMP     (R3),R2
16549  063722  031302                  BIT     (R3),R2
16550  063724  041302                  BIC     (R3),R2
16551  063726  051302                  BIS     (R3),R2
16552  063730  006702                  SXT     R2
16553  063732  005402                  NEG     R2
16554  063734  161302                  SUB     (R3),R2
16555  063736  020312                  CMP     R3,(R2)
16556  063740  030312                  BIT     R3,(R2)
16557  063742  120312                  CMPB    R3,(R2)
16558  063744  131302                  BITB    (R3),R2
16559  063746  005712                  TST     (R2)
16560  063750  105712                  TSTB    (R2)
16561  063752  021312                  CMP     (R3),(R2)
16562  063754  031312                  BIT     (R3),(R2)
16563  063756  121312                  CMPB    (R3),(R2)
16564  063760  131312                  BITB    (R3),(R2)
16565  063762  061302                  ADD     (R3),R2
16566  063764  000302                  SWAB    R2
16567  063766  160302                  SUB     R3,R2
```

```
16568  063770  060302                  ADD     R3,R2
16569  063772  010302                  MOV     R3,R2
16570  063774  011302                  MOV     (R3),R2
16571  063776  110302                  MOVB    R3,R2
16572  064000  006102                  ROL     R2
16573  064002  106102                  ROLB    R2
16574  064004  105402                  NEGB    R2
16575  064006  102400                  BVS     .+2
16576  064010  102000                  BVC     .+2
16577  064012  000005                  RESET
16578  064014  020302                  CMP     R3,R2
16579  064016  030302                  BIT     R3,R2
16580  064020  040302                  BIC     R3,R2
16581  064022  120302                  CMPB    R3,R2
16582  064024  130302                  BITB    R3,R2
16583  064026  140302                  BICB    R3,R2
16584  064030  150302                  BISB    R3,R2
16585
16586  064032  000000          DWTB:   0                       ;ALSO SERVES AS INSTAB TABLE TERMINATOR
16587  064034  000001                  1
16588  064036  000400                  400
16589  064040  177401                  177401
16590  064042  052526                  52526
16591  064044  125253                  125253
16592                           ;*      MED TEST TABLES
16593
16594  064046  000000          TLOC1:  .WORD   0
16595  064050  000000          PSWHOL: .WORD   0
16596  064052  000000          TABBEG: .WORD   0
16597  064054  000000          TABEND: .WORD   0
16598  064056  000040          STGBLK: .BLKW   40
16599  064156  000000          VADR:   .WORD   0
16600  064160  000000          PA1716: .WORD   0
16601  064162  000000          PA1500: .WORD   0
16602  064164  000000          TLOC2:  .WORD   0
16603
16604                           ;*
16605                           ;*      TABLE II
16606                           ;*
16607                           ;*      FOLLOWING IS A TABLE OF INTERNAL REGISTER OPERATION CODES
16608                           ;*      USED FOR TESTING THE MED INSTRUCTION.  LABELS CORRESPOND
16609                           ;*      TO REGISTER NAMES, THE HIGH BYTE IS THE READ OPERATION
16610                           ;*      CODE, THE LOW BYTE THE WRITE CODE.
16611                           ;*              NOTE:  WHEN ADDING OR DELETING
16612                           ;*              ENTRIES IN THIS TABLE, CHECK DUAL
16613                           ;*              ADDRESING TEST TO SEE THAT THE "SCRATCH
16614                           ;*              PAD LIMITS" ARE MAINTAINED.
16615                           ;*
16616
16617  064166          TBL2:
16618
16619  064166          ASP1:                           ;A SCRATCH PAD - LO
16620  064166  201    001      R1A:    .BYTE   201,001 ;LOBYTE, HIBYTE=WRITE CODE, READ CODE
16621  064170  202    002      R2A:    .BYTE   202,002
16622  064172  203    003      R3A:    .BYTE   203,003
16623  064174  204    004      R4A:    .BYTE   204,004
```

```
16624  064176      205      005          R5A:     .BYTE   205,005
16625  064200      206      006          R6A:     .BYTE   206,006
16626  064202      210      010          FAC3.0:  .BYTE   210,010
16627  064204      211      011          FAC3.1:  .BYTE   211,011
16628  064206      212      012          FAC3.2:  .BYTE   212,012
16629  064210      213      013          FAC3.3:  .BYTE   213,013
16630  064212      214      014          FAC3.4:  .BYTE   214,014
16631  064214      215      015          FAC3.5:  .BYTE   215,015
16632  064216      216      016          UR6A:    .BYTE   216,016
16633  064220      217      017          FDST3:   .BYTE   217,017
16634  064222      220      020          WCSA.0:  .BYTE   220,020          ;A SCRATCH PAD-HI
16635  064224      221      021          WCSA.1:  .BYTE   221,021
16636  064226      222      022          GNWHAM:  .BYTE   222,022
16637  064230      223      023          CNSTSW:  .BYTE   223,023
16638  064232      226      026          CNSSW:   .BYTE   226,026
16639  064234      227      027          CNSCDR:  .BYTE   227,027
16640  064236      230      030          FAC1.0:  .BYTE   230,030
16641  064240      231      031          FAC1.1:  .BYTE   231,031
16642  064242      232      032          FAC1.2:  .BYTE   232,032
16643  064244      233      033          FAC1.3:  .BYTE   233,033
16644  064246      234      034          FAC1.4:  .BYTE   234,034
16645  064250      235      035          FAC1.5:  .BYTE   235,035
16646  064252      236      036          FPSHI:   .BYTE   236,036
16647  064254      237      037          ASP2:    FDST1:  .BYTE   237,037
16648
16649  064256                            BSP1:
16650  064256      241      041          R1B:     .BYTE   241,041          ;B SCRATCH PAD - LO
16651  064260      242      042          R2B:     .BYTE   242,042
16652  064262      243      043          R3B:     .BYTE   243,043
16653  064264      244      044          R4B:     .BYTE   244,044
16654  064266      245      045          R5B:     .BYTE   245,045
16655  064270      246      046          R6B:     .BYTE   246,046
16656  064272      250      050          FAC2.0:  .BYTE   250,050
16657  064274      251      051          FAC2.1:  .BYTE   251,051
16658  064276      252      052          FAC2.2:  .BYTE   252,052
16659  064300      253      053          FAC2.3:  .BYTE   253,053
16660  064302      254      054          FAC2.4:  .BYTE   254,054
16661  064304      255      055          FAC2.5:  .BYTE   255,055
16662  064306      256      056          UR6B:    .BYTE   256,056
16663  064310      257      057          FDST2:   .BYTE   257,057
16664  064312      260      060          WCSB.0:  .BYTE   260,060          ;B SCRATCH PAD - HI
16665  064314      261      061          WCSB.1:  .BYTE   261,061
16666  064316      262      062          WCSADR:  .BYTE   262,062
16667  064320      263      063          RZERO:   .BYTE   263,063
16668  064322      266      066          RVECT:   .BYTE   266,066
16669  064324      270      070          FAC0.0:  .BYTE   270,070
16670  064326      272      072          FAC0.1:  .BYTE   272,072
16671  064330      273      073          FAC0.2:  .BYTE   273,073
16672  064332      274      074          FAC0.4:  .BYTE   274,074
16673  064334      275      075          FAC0.5:  .BYTE   275,075
16674  064336      276      076          FEA:     .BYTE   276,076
16675  064340      277      077          BSP2:    FDST0:  .BYTE   277,077
16676
16677  064342                            CSP1:
16678  064342      300      100          LJAM:    .BYTE   300,100          ;C SCRATCH PAD
16679  064344      301      101          LSERV:   .BYTE   301,101
```

```
16680  064346  302  102  LPBA:   .BYTE   302,102
16681  064350  303  103  LCUA:   .BYTE   303,103
16682  064352  304  104  LFGIN:  .BYTE   304,104
16683  064354  305  105  LWHAM:  .BYTE   305,105
16684  064356  307  107  LTAG:   .BYTE   307,107
16685  064360  310  110  CNSC0:  .BYTE   310,110
16686  064362  311  111  CNSC1:  .BYTE   311,111
16687  064364  312  112  CNSC2:  .BYTE   312,112
16688  064366  313  113  CST200: .BYTE   313,113
16689  064370  316  116  CSP2:   CNST0:  .BYTE   316,116
16690  064372  224  024  RT1A:   .BYTE   224,024
16691  064374  225  025  RT2A:   .BYTE   225,025
16692  064376  264  064  RT1B:   .BYTE   264,064
16693  064400  265  065  RT2B:   .BYTE   265,065
16694  064402  000000       .WORD   0
16695
16696                       ;*
16697                       ;*      TABLE III
16698                       ;*
16699                       ;*      THE FOLLOWING IS A LIST OF "NOP" OPERATION CODES
16700                       ;*      THAT WILL BE USED WITH A MED IN MED TEST 3 TO
16701                       ;*      ENSURE THAT A MED WITH THESE CODES WILL NOT HANG.
16702
16703  064404               TBL3:
16704  064404  120  137  NOPS:   .BYTE   120,137          ;GROUP A
16705  064406  145  145       .BYTE   145,145          ;GROUP B
16706  064410  150  151       .BYTE   150,151          ;GROUP C
16707  064412  156  177       .BYTE   156,177          ;GROUP D
16708  064414  320  343       .BYTE   320,343          ;GROUP E
16709  064416  353  357       .BYTE   353,357          ;FROUP G
16710  064420  000000       .WORD   0                ;A 0 TERMINATES TABLE
16711
16712                       ;*
16713                       ;*      TABLE IV
16714                       ;*
16715                       ;*      THE LIST BELOW CONTAINS THOSE OPERATION CODES
16716                       ;*      CORRESPONDING TO THE INTERNAL REGISTERS WHICH MUST
16717                       ;*      BE TESTED SEPERATELY BECAUSE THEY ARE READ-ONLY,
16718                       ;*      WRITE-ONLY, OR USED IN MACRO CODE EXECUTION, ETC. . .
16719
16720  064422               TBL4:
16721  064422  200  000  R0A:    .BYTE   200,000          ;LOBYTE,HYBYTE - WRITE CODE, READ CODE
16722  064424  207  007  R7A:    .BYTE   207,007          ;0 REPLACES ANY NON EXSISTENT CODES
16723  064426  240  040  R0B:    .BYTE   240,040          ;EXCEPT IN THE CASE OF R0A
16724  064430  247  047  R7B:    .BYTE   247,047
16725  064432  314  114  CNST2:  .BYTE   314,114
16726  064434  317  117  CNST1:  .BYTE   317,117
16727                       ;*      TABLE V
16728                       ;*
16729  064436               TBL5:
16730
16731  064436  306          LCDTA:  .BYTE   306              ;THIS TABLE CONTAINS THE OPERATION
16732  064437  106                  .BYTE   106              ;CODES OF THOSE INTERNAL REGISTERS
16733  064440  315          MD:     .BYTE   315              ;WHICH MUST BE TESTED USING THE
16734  064441  115                  .BYTE   115              ;MICROBREAK REGISTER.  THEIR
16735  064442  267          CNSCTL: .BYTE   267              ;ASSOCIATED MICRO-ADDRESSES ARE IN
```

```
16736  064443  067                     .BYTE   067             ;THE NEXT TABLE
16737  064444  140             JAM.    .BYTE   140
16738  064445  141             SERV:   .BYTE   141
16739  064446  142             PBA:    .BYTE   142
16740  064447  143             CUA:    .BYTE   143
16741  064450  344             FLAG:   .BYTE   344
16742  064451  144                     .BYTE   144
16743  064452  345             DREG:   .BYTE   345
16744  064453  146             REV:    .BYTE   146
16745  064454  346             SREG:   .BYTE   346
16746  064455  147             COUNT:  .BYTE   147
16747  064456  347             NUA:    .BYTE   347
16748  064457  351             RES:    .BYTE   351
16749  064460  152             DCS0:   .BYTE   152
16750  064461  352                     .BYTE   352             ;INIT REG
16751  064462  153             DCS1:   .BYTE   153
16752  064463  000                     .BYTE   0               ;TABLE TERMINATOR
16753                                   .EVEN
16754
16755                          ;*       TABLE VI
16756                          ;*
16757  064464                  TBL6:
16758
16759  064464  003330          ULCDTA: .WORD   3330            ;THIS TABLE CONTAINS THE MICRO-ADDRESSES
16760  064466  003150                  .WORD   3150            ;WHICH ARE LOADED INTO THE MICROBREAK
16761  064470  003375          UMD:    .WORD   3375            ;REG. TO TEST THE OPERATION CODES
16762  064472  003271                  .WORD   3271            ;CONTAINED IN THE PRECEEDING TABLE.
16763  064474  003240          UCNSCTL:.WORD   3240
16764  064476  003224                  .WORD   3224
16765  064500  003160          UJAM:   .WORD   3160
16766  064502  003161          USERV:  .WORD   3161
16767  064504  003170          UPBA:   .WORD   3170
16768  064506  003171          UCUA:   .WORD   3171
16769  064510  003344          UFLAG:  .WORD   3344
16770  064512  003320                  .WORD   3320
16771  064514  003345          UDREG:  .WORD   3345
16772  064516  003340          UREV:   .WORD   3340
16773  064520  003350          USREG:  .WORD   3350
16774  064522  003341          UCOUNT: .WORD   3341
16775  064524  003351          UNUA:   .WORD   3351
16776  064526  003355          URES:   .WORD   3355
16777  064530  003720          UDCS0:  .WORD   3720
16778  064532  003724          UINIT:  .WORD   3724
16779  064534  003721          UDCS1:  .WORD   3721
16780
16781                          ;*       TABLE VII
16782                          ;*
16783                          ;*       THIS TABLE HOLDS THE OPERATION CODES AND THE CONSTANT
16784                          ;*       VALUE EXPECTED FOR CERTAIN INTERNAL REGISTERS.
16785  064536                  TBL7:
16786
16787  064536  000100  077600  CLJAM:  .WORD   100,77600
16788  064542  000101  000010  CLSERV: .WORD   101,10
16789  064546  000102  020000  CLPBA:  .WORD   102,20000
16790  064552  000103  000004  CLCUA:  .WORD   103,4
16791  064556  000104  050000  CLFGIN: .WORD   104,50000
```

```
16792  064562  000105  054000          CLWHAM: .WORD   105,54000
16793  064566  000107  024000          CLTAG:  .WORD   107,24000
16794  064572  000110  177400          CCNSC0: .WORD   110,177400
16795  064576  000111  177600          CCNSC1: .WORD   111,177600
16796  064602  000112  100000          CCNSC2: .WORD   112,100000
16797  064606  000113  000200          CCST200:.WORD   113,200
16798  064612  000114  000002          CCNST2: .WORD   114,2
16799  064616  000116  000000          CCNST0: .WORD   116,0
16800  064622  000117  000001          CCNST1: .WORD   117,1
16801  064626  000000                          .WORD   0
16802
16803                                  .EVEN
16804  064630                          DBTA:
16805  064630      000     377     252 .BYTE   000,377,252,125
16806  064633      125
16807  064634                          DBTB:
16808  064634      000     001     120 .BYTE   000,001,120,253
16809  064637      253
16810
16811                                  ;MESSAGE TABLES
16812
16813  064640                          EM1:
16814  064640                          EM2:
16815  064640  027523  020102  051504  EM4:    .ASCII  'S/B DST '
16816  064646  020124
16817  064650  040527  020123  051504  EM7:    .ASCII  'WAS DST '
16818  064656  020124
16819  064660  042040  051505  004524  EM6:    .ASCII  ' DEST'<HT>
16820  064666  024040  051111  004451  EM5:    .ASCIZ  ' (IR)'<HT>' TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16821  064674  052040  051505  004524
16822  064702  024040  041520  004451
16823  064710  024040  050123  004451
16824  064716  050050  053523  000051
16825  064724  027523  020102  042522  EM10:   .ASCIZ  'S/B RES WAS RES DST OP  SRC OP  TEST'<HT>' (PC)'<HT>' (SP)'<HT>'(PSW)'
16826  064732  020123  040527  020123
16827  064740  042522  020123  051504
16828  064746  020124  050117  020040
16829  064754  051123  020103  050117
16830  064762  020040  042524  052123
16831  064770  020011  050050  024503
16832  064776  020011  051450  024520
16833  065004  024011  051520  024527
16834  065012      000
16835  065013      123 041057  051440  EM3:    .ASCIZ  'S/B SP'<HT>'WAS SP'<HT>' (IR)'<HT>' TEST'<HT>' (PC)'<HT>'(PSW)'
16836  065020  004520  040527  020123
16837  065026  050123  020011  044450
16838  065034  024522  020011  042524
16839  065042  052123  020011  050050
16840  065050  024503  024011  051520
16841  065056  024527      000
16842  065061      011 020011  051511  DH2:    .ASCIZ  <HT><HT>' IS R3'
16843  065066  051040  000063
16844  065072  004411  044440  020123  DH4:    .ASCIZ  <HT><HT>' IS R5'
16845  065100  032522      000
16846  065103      015 042412  042116  EOP1:   .ASCIZ  <15><12>'END PASS # '
16847  065110  050040  051501  020123
```

```
16848  065116  020043  000
16849  065121  011  051105  047522  EOP2:   .ASCIZ <HT>'ERROR COUNT = '
16850  065126  020122  047503  047125
16851  065134  020124  020075  000
16852  065141  015  041412  045521  IDENT1: .ASCIZ <15><12>'CQYDADO KD11-K BASIC LOGIC TESTS'<15><12>
16853  065146  040504  030104  045440
16854  065154  030504  026461  020113
16855  065162  040502  044523  020103
16856  065170  047514  044507  020103
16857  065176  042524  052123  006523
16858  065204  000012
16859  065206  005015  051124  050101  BEMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 4 PC = '
16860  065214  042520  020104  047524
16861  065222  032040  050040  020103
16862  065230  020075  000
16863  065233  015  052012  040522  RSMSG:  .ASCIZ  <CR><LF>'TRAPPED TO 10 PC = '
16864  065240  050120  042105  052040
16865  065246  020117  030061  050040
16866  065254  020103  020075  000
16867  065261  124  051505  051524  EM11:   .ASCIZ  'TESTS SKIPPED'
16868  065266  051440  044513  050120
16869  065274  042105  000
16870  065277  040  050040  004503  DH11:   .ASCIZ  '  PC''<HT>''EXPCTD''<HT>''ACTUAL''<HT>''(TEST #'S)''
16871  065304  054105  041520  042124
16872  065312  040411  052103  040525
16873  065320  004514  052050  051505
16874  065326  020124  023443  024523
16875  065334  000
16876  065335  115  042105  042040  EM12:   .ASCIZ  /MED DID NOT ABORT IN USER MODE/
16877  065342  042111  047040  052117
16878  065350  040440  047502  052122
16879  065356  044440  020116  051525
16880  065364  051105  046440  042117
16881  065372  000105
16882  065374  042515  020104  054105  EM13:   .ASCIZ  /MED EXECUTED IN USER MODE/
16883  065402  041505  052125  042105
16884  065410  044440  020116  051525
16885  065416  051105  046440  042117
16886  065424  000105
16887  065426  042515  020104  044103  EM14:   .ASCIZ  /MED CHANGED PSW/
16888  065434  047101  042507  020104
16889  065442  051520  000127
16890  065446  044515  051103  041117  EM15:   .ASCIZ  /MICROBREAK TRAP-TO-4 DID NOT OCCUR/
16891  065454  042522  045501  052040
16892  065462  040522  026520  047524
16893  065470  032055  042040  042111
16894  065476  047040  052117  047440
16895  065504  041503  051125  000
16896  065511  114  043517  052503  EM17:   .ASCIZ  /LOGCUA LOGGED WRONG/
16897  065516  020101  047514  043507
16898  065524  042105  053440  047522
16899  065532  043516  000
16900  065535  103  050123  041440  EM21:   .ASCIZ  /CSP CONSTANT WRONG/
16901  065542  047117  052123  047101
16902  065550  020124  051127  047117
16903  065556  000107
```

```
16904  065560  040502  020104  040504  EM22:   .ASCIZ  /BAD DATA READ BY A MED/
16905  065566  040524  051040  040505
16906  065574  020104  054502  040440
16907  065602  046440  042105     000
16908  065607     116  020117  042117  EM23:   .ASCIZ  /NO ODD PC TRAP/
16909  065614  020104  041520  052040
16910  065622  040522  000120
16911  065626  042117  020104  042101  EM24:   .ASCIZ  /ODD ADR. BIT NOT SET IN CPU ERR REG OR LOG JAM/
16912  065634  027122  041040  052111
16913  065642  047040  052117  051440
16914  065650  052105  044440  020116
16915  065656  050103  020125  051105
16916  065664  020122  042522  020107
16917  065672  051117  046040  043517
16918  065700  045040  046501     000
16919  065705     120  054510  020123  EM26:   .ASCIZ  /PHYS BA LOGGED WRONG/
16920  065712  040502  046040  043517
16921  065720  042507  020104  051127
16922  065726  047117  000107
16923  065732  040503  044103  020105  EM27:   .ASCIZ  /CACHE PARITY ERROR LOGGED IN BAKUP MODE/
16924  065740  040520  044522  054524
16925  065746  042440  051122  051117
16926  065754  046040  043517  042507
16927  065762  020104  047111  041040
16928  065770  045501  050125  046440
16929  065776  042117  000105
16930  066002  040503  044103  020105  EM30:   .ASCIZ  /CACHE PARITY TRAPPED WHEN DISABLED/
16931  066010  040520  044522  054524
16932  066016  052040  040522  050120
16933  066024  042105  053440  042510
16934  066032  020116  044504  040523
16935  066040  046102  042105     000
16936  066045     111  051516  051124  EM41:   .ASCIZ  /INSTR. NOT ABORTED IN CACHE ABURT MODE/
16937  066052  020056  047516  020124
16938  066060  041101  051117  042524
16939  066066  020104  047111  041440
16940  066074  041501  042510  040440
16941  066102  047502  052122  046440
16942  066110  042117  000105
16943  066114  042515  047515  054522  EM32:   .ASCIZ  /MEMORY ERP REG INCORRECT/
16944  066122  042440  051122  051040
16945  066130  043505  044440  041516
16946  066136  051117  042522  052103
16947  066144     000
16948  066145     124  046511  047505  EM33:   .ASCIZ  /TIMEOUT BIT NOT SET IN CPU ERR REG OR LOG JAM/
16949  066152  052125  041040  052111
16950  066160  047040  052117  051440
16951  066166  052105  044440  020116
16952  066174  050103  020125  051105
16953  066202  020122  042522  020107
16954  066210  051117  046040  043517
16955  066216  045040  046501     000
16956  066223     116  020117  046111  EM34:   .ASCIZ  /NO ILLEGAL INTERNAL ADR TRAP/
16957  066230  042514  040507  020114
16958  066236  047111  042524  047122
16959  066244  046101  040440  051104
```

```
16960  066252  052040  040522  000120
16961  066260  047111  051124  040516  EM35:   .ASCIZ  /INTRNAL ADR ERR BIT NOT SET IN CPU ERR REG OR LOG JAM/
16962  066266  020114  042101  020122
16963  066274  051105  020122  044502
16964  066302  020124  047516  020124
16965  066310  042523  020124  047111
16966  066316  041440  052520  042440
16967  066324  051122  051040  043505
16968  066332  047440  020122  047514
16969  066340  020107  040512  000115
16970  066346  040514  052123  044440  EM36:   .ASCIZ  "LAST INTR/TRAP VECTOR NOT LOGGED IN FLAG REG"
16971  066354  052116  027522  051124
16972  066362  050101  053040  041505
16973  066370  047524  020122  047516
16974  066376  020124  047514  043507
16975  066404  042105  044440  020116
16976  066412  046106  043501  051040
16977  066420  043505     000
16978  066423     114  043517  043040  EM37:   .ASCIZ  /LOG FIRST MODE DID NOT INHIBIT ERROR LOG AFTER FIRST ERROR/
16979  066430  051111  052123  046440
16980  066436  042117  020105  044504
16981  066444  020104  047516  020124
16982  066452  047111  044510  044502
16983  066460  020124  051105  047522
16984  066466  020122  047514  020107
16985  066474  043101  042524  020122
16986  066502  044506  051522  020124
16987  066510  051105  047522  000122
16988  066516  051105  047522  020122  EM40:   .ASCIZ  /ERROR LOG WAS NOT REENABLED, ODD ADR BIT CLR IN CPUERR/
16989  066524  047514  020107  040527
16990  066532  020123  047516  020124
16991  066540  042522  047105  041101
16992  066546  042514  026104  047440
16993  066554  042104  040440  051104
16994  066562  041040  052111  041440
16995  066570  051114  044440  020116
16996  066576  050103  042525  051122
16997  066604     000
16998  066605     116  020117  040503  EM31:   .ASCIZ  /NO CACHE PARITY TRAP/
16999  066612  044103  020105  040520
17000  066620  044522  054524  052040
17001  066626  040522  000120
17002  066632  047514  023040  044040  EM42:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN LOG SERVICE/
17003  066640  020111  054502  042524
17004  066646  023040  052040  043501
17005  066654  050040  051101  052111
17006  066662  020131  044502  051524
17007  066670  047040  052117  051440
17008  066676  052105  044440  020116
17009  066704  047514  020107  042523
17010  066712  053122  041511  000105
17011  066720  047514  023040  044040  EM43:   .ASCIZ  /LO & HI BYTE & TAG PARITY BITS NOT SET IN MEM ERR REG/
17012  066726  020111  054502  042524
17013  066734  023040  052040  043501
17014  066742  050040  051101  052111
17015  066750  020131  044502  051524
```

```
17016  066756  047040  052117  051440
17017  066764  052105  044440  020116
17018  066772  042515  020115  051105
17019  067000  020122  042522  000107
17020  067006  040503  044103  020105  EM45:   .ASCIZ  /CACHE TAG LOGGED WRONG/
17021  067014  040524  020107  047514
17022  067022  043507  042105  053440
17023  067030  047522  043516     000
17024  067035     103  041501  042510  EM16:   .ASCIZ  /CACHE DATA LOGGED WRONG/
17025  067042  042040  052101  020101
17026  067050  047514  043507  042105
17027  067056  053440  047522  043516
17028  067064     000
17029  067065     105  051511  051440  EMEIS1: .ASCIZ  'EIS SET COND CODES WRONG'
17030  067072  052105  041440  047117
17031  067100  020104  047503  042504
17032  067106  020123  051127  047117
17033  067114  000107
17034  067116  044505  020123  040507  EMEIS2: .ASCIZ  'EIS GAVE WRONG RESULT'
17035  067124  042526  053440  047522
17036  067132  043516  051040  051505
17037  067140  046125  000124
17038  067144  052501  047524  044455  EM46:   .ASCIZ  'AUTO-INCREMENT (DECREMT) DID NOT OCCUR IN EIS'
17039  067152  041516  042522  042515
17040  067160  052116  024040  042504
17041  067166  051103  046505  024524
17042  067174  042040  042111  047040
17043  067202  052117  047440  041503
17044  067210  051125  044440  020116
17045  067216  044505  000123
17046  067222  050040  053523  051011  DHEIS1: .ASCII  ' PSW'<HT>'REG-WAS-REG+1'<HT>'REG-S/B-REG+1'<HT>
17047  067230  043505  053455  051501
17048  067236  051055  043505  030453
17049  067244  051011  043505  051455
17050  067252  041057  051055  043505
17051  067260  030453     011
17052  067263     040  050040  004503  DH46:   .ASCIZ  '  PC'<HT>' (IR)'<HT>' TEST'
17053  067270  024040  051111  004451
17054  067276  052040  051505  000124
17055  067304  020040  041520  046411  DH15:   .ASCIZ  /  PC/<HT>/MEDCODE MICROBK REG./
17056  067312  042105  047503  042504
17057  067320  046440  041511  047522
17058  067326  045502  051040  043505
17059  067334  000056
17060  067336  020040  041520  046411  DH17:   .ASCIZ  /  PC/<HT>/MEDCODE EXPECTD RECEIVD/
17061  067344  042105  047503  042504
17062  067352  042440  050130  041505
17063  067360  042124  051040  041505
17064  067366  044505  042126     000
17065  067373     040  050040  000103  DH23:   .ASCIZ  /  PC/
17066  067400  020040  041520  041411  DH24:   .ASCIZ  /  PC/<HT>/CPUERR/<HT>/LOGJAM/
17067  067406  052520  051105  004522
17068  067414  047514  045107  046501
17069  067422     000
17070  067423     040  050040  004503  DH25:   .ASCIZ  /  PC/<HT>/FLGREG/
17071  067430  046106  051107  043505
```

```
17072  067436      000
17073  067437      040  050040  004503  DH26:   .ASCIZ  '  PC'<HT>'<17:16>-S/B PA-<15:0>  <17:16>-WAS PA-<15:0>'
17074  067444  030474  035067  033061
17075  067452  026476  027523  020102
17076  067460  040520  036055  032461
17077  067466  030072  020076  036040
17078  067474  033461  030472  037066
17079  067502  053455  051501  050040
17080  067510  026501  030474  035065
17081  067516  037060      000
17082  067521      040  050040  004503  DH27:   .ASCIZ  /  PC/<HT>/LOGPBA/<HT>/LOGDATA/<HT>/LOGTAG/
17083  067526  047514  050107  040502
17084  067534  046011  043517  040504
17085  067542  040524  046011  043517
17086  067550  040524  000107
17087  067554  020040  041520  046411  DH32:   .ASCIZ  /  PC/<HT>/MEMERR/
17088  067562  046505  051105  000122
17089  067570  020040  041520  046011  DH42:   .ASCIZ  /  PC/<HT>/LOGSERVCE/
17090  067576  043517  042523  053122
17091  067604  042503      000
17092  067607      040  050040  004503  DH44:   .ASCIZ  /  PC/<HT>/EXPCT/<HT>/RECVD/
17093  067614  054105  041520  004524
17094  067622  042522  053103  000104
17095                           .EVEN
17096  067630  001016  001076  001100  DT15:   .WORD   $ERRPC,$TMP0,$TMP1,0
17097  067636  000000
17098  067640  001016  001100  001102  DT21:   .WORD   $ERRPC,$TMP1,$TMP2,$REG0,0
17099  067646  001062  000000
17100  067652  001016  001100  001102  DT22:   .WORD   $ERRPC,$TMP1,$TMP2,$TMP3,0
17101  067660  001104  000000
17102  067664  001016  000000          DT23:   .WORD   $ERRPC,0
17103  067670  001016  001064  001062  DT24:   .WORD   $ERRPC,$REG1,$REG0,0
17104  067676  000000
17105  067700  001016  001062  000000  DT25:   .WORD   $ERRPC,$REG0,0
17106  067706  001016  001064  001066  DT26:   .WORD   $ERRPC,$REG1,$REG2,$REG0,$REG3,0
17107  067714  001062  001070  000000
17108  067722  001016  001070  001064  DT27:   .WORD   $ERRPC,$REG3,$REC1,$REG2,0
17109  067730  001066  000000
17110  067734  001060  001066  001070  DTEIS1: .WORD   $REGAD,$REG2,$REG3,$REG1,$REG4
17111  067742  001064  001072
17112  067746  001016  001076  001062  DT46:   .WORD   $ERRPC,$TMP0,$REG0,0
17113  067754  000000
17114
17115  067756      000      000        DF15:   .BYTE   0,0
17116  067760      000      000   000  DF17:   .BYTE   0.0,0
17117          067764                  .EVEN
17118  067764                          DT1:
17119  067764                          DT2:
17120  067764                          DT4:
17121  067764  001072                  DT10:   .WORD   $REG4
17122  067766  001070                  DT7:    .WORD   $REG3
17123  067770  001066                  DT6:    .WORD   $REG2
17124  067772  001064  001062  001016  DT5:    .WORD   $REG1,$REG0,$ERRPC,$REG5,$REGAD,0
17125  070000  001074  001060  000000
17126  070006  001072  001070  001064  DT3:    .WORD   $REG4,$REG3,$REG1,$REG0,$ERRPC,$REGAD,0
17127  070014  001062  001016  001060
```

```
17128  070022  000000
17129  070024  001016  001124  001062  DT11:   .WORD   $ERRPC,$TESTN,$REG0,0
17130  070032  000000
17131          000001                  .END
```

```
ABASE = 000000            930
ACDW1 = 000000            930
ACDW2 = 000000            930
ACPUOP= 000000            930     945
ADDW0 = 000000            930
ADDW1 = 000000            930
ADDW10= 000000            930
ADDW11= 000000            930
ADDW12= 000000            930
ADDW13= 000000            930
ADDW14= 000000            930
ADDW15= 000000            930
ADDW2 = 000000            930
ADDW3 = 000000            930
ADDW4 = 000000            930
ADDW5 = 000000            930
ADDW6 = 000000            930
ADDW7 = 000000            930
ADDW8 = 000000            930
ADDW9 = 000000            930
ADEVCT= 000000            930     936
ADEVM = 000000            930
AENV  = 000000            930     941
AENVM = 000000            930     942
AFATAL= 000000            930     933
ALUADD  063334          14004   14009   16409#
ALUSUB  063554          14064   14069   16493#
AMADR1= 000000            930
AMADR2= 000000            930
AMADR3= 000000            930
AMADR4= 000000            930
AMAMS1= 000000            930
AMAMS2= 000000            930
AMAMS3= 000000            930
AMAMS4= 000000            930
AMSGAD= 000000            930     938
AMSGLG= 000000            930     939
AMSGTY= 000000            930     932
AMTYP1= 000000            930
AMTYP2= 000000            930
AMTYP3= 000000            930
AMTYP4= 000000            930
ANDTAB  063414          14124   14129   16437#
APASS = 000000            930     935
APRIOR= 000000            930
APTCSU= 000040          16147   16331#
APTENV= 000001          16036   16140   16287   16329#
APTSIZ= 000200          16328#
APTSPO= 000100          16142   16289   16330#
ASP1    064166          16619#
ASP2    064254          16647#
ASWREG= 000000            930     943
ATA     063276           2638    2688    2700    2716    2748    2761    2809    2830    2849    2861    2871    2883    2894
                         7629    7640    7653    7664    7677    7689    7702    7713    7726    7738    7751    7770    7791
                         7812    7833    7862    7891    7892    7906    7920    7921    7935    7949    7978    8007    8008
                         8022    8036    8037    8051    8065    8087    8109    8131    8769    8781    8821    8833    8868
```

```
                    9093   9107   9130   9144   9245   9259   9282   9296   9383   9415   9487   9529   9764
                    9769   9818   9823   9867   9932  11081  11124  11168  14406  14426  14478  14498  14545
                   16393#
ATESTN= 000000       930    934
AUNIT = 000000       930    937
AUSWR = 000000       930    944
AVECT1= 000000       930
AVECT2= 000000       930
A002    001636      1282#  1286
A015    002106      1448   1453#
A017    002200      1491   1498#
A020    002246      1520   1525#
A021    002276      1542   1547#
A023    002364      1583   1588#
A032    002570      1688   1693#
A033    002630      1710   1715#
A034    002670      1731   1736#
A041    003034      1812   1817#
A046    003234      1910   1915#
BASIC   003246      1916   1924#
BEMSG   065206     15907  16859#
BERFLG  063250      4465*  4471*  15898*  16380#
BERR    061220      4503  12312   12316   13180   13218   13257   13274   13326   13386   13424   13455   13487   14914
                   15141  15167   15333   15363   15411   15539   15586   15617   15633   15652   15667   15901#
BETST   061212      4462  15898#
BIT0  = 000001       812#   1219    2294   15158   15415   15452   15533   15570   15607   15634
BIT00 = 000001       802#    812
BIT01 = 000002       801#    811
BIT02 = 000004       800#    810
BIT03 = 000010       799#    809
BIT04 = 000020       798#    808
BIT05 = 000040       797#    807
BIT06 = 000100       796#    806
BIT07 = 000200       795#    805
BIT08 = 000400       794#    804
BIT09 = 001000       793#    803   15961   16046
BIT1  = 000002       811#   2955   15483
BIT10 = 002000       792#  10023
BIT11 = 004000       791#  10893   15039   15968
BIT12 = 010000       790#  11705
BIT13 = 020000       789#  12759   15344   16031
BIT14 = 040000       788#  13899   15943
BIT15 = 100000       787#  14850   15089   15116   15158   15177   15452   15493   15570   15600   15607   15845
BIT2  = 000004       810#   3687   15177   15375   15600
BIT3  = 000010       809#   4196
BIT4  = 000020       808#   4921   15337   15590
BIT5  = 000040       807#   1223    5794   15421
BIT6  = 000100       806#   1218    1221    6646   15171   15367   15618
BIT7  = 000200       805#   1220    1222    7502   15344   15520
BIT8  = 000400       804#   8317
BIT9  = 001000       803#   9238   15084   15135   15344
BKFLAG  061112     15081*  15093   15113*  15123   15788*  15791#
BKROUT  061104     15079   15788#
BPTLOC  063234      2294    2955    3687    4196    4921    5794    6646    7502    8317    9238   10023   10893   11705
                   12759   13899   14850   16373#
BPTVEC= 000014       819#
```

```
BSP1    064256          16649#
BSP2    064340          16675#
BT001   001630           1272#
BT002   001634           1272    1280#
BT003   001646           1282    1294#   1301
BT004   001662           1297    1307#   1316
BT005   001700           1312    1322#   1331
BT006   001716           1327    1337#   1344
BT007   001730           1341    1350#   1359
BT010   001744           1355    1365#   1374
BT011   001762           1371    1380#   1390
BT012   002002           1387    1396#   1405
BT013   002022           1402    1411#   1422
BT014   002046           1419    1428#   1437
BT015   002070           1434    1443#   1451    1458
BT016   002116           1455    1464#   1476
BT017   002144           1472    1482#   1496    1505
BT020   002220           1500    1511#   1523    1529
BT021   002256           1526    1535#   1545    1552
BT022   002310           1549    1558#   1568
BT023   002334           1565    1574#   1586    1592
BT024   002374           1589    1598#   1607
BT025   002420           1604    1613#   1621
BT026   002440           1618    1626#   1634
BT027   002460           1631    1640#   1648
BT030   002500           1645    1654#   1662
BT031   002520           1659    1667#   1675
BT032   002540           1672    1681#   1691    1697
BT033   002600           1694    1703#   1713    1719
BT034   002640           1716    1724#   1734    1740
BT035   002700           1737    1745#   1754
BT036   002720           1751    1759#   1768
BT037   002740           1765    1773#   1782
BT040   002760           1779    1788#   1799
BT041   003006           1796    1804#   1815    1821
BT042   003044           1818    1827#   1837
BT043   003064           1834    1843#   1854
BT044   003106           1851    1860#
BT045   003134           1867    1870    1875#   1885
BT046   003162           1882    1891#   1913    1919
CATERR  063252          15833   15835*  15849*  15901   15903*  15910*  16381#
CCNSC0  064572          16794#
CCNSC1  064576          16795#
CCNSC2  064602          16796#
CCNST0  064616          16799#
CCNST1  064622          16800#
CCNST2  064612          16798#
CCR   = 177746           1217#  15222*  15224*  15274*  15288*  15290*  15405*  15406   15445*  15449*  15451*  15455*  15470*
CCST20  064606          16797#
CITST   013412           4606    4618#
CLCUA   064552          16790#
CLFGIN  064556          16791#
CLJAM   064536          16787#
CLPBA   064546          16789#
CLSERV  064542          16788#
CLTAG   064566          16793#
```

```
CLWHAM  064562          16792#
CNSCDR  064234          16639#
CNSCTL  064442          16735#
CNSC0   064360          16685#
CNSC1   064362          16686#
CNSC2   064364          16687#
CNSSW   064232          16638#
CNSTSW  064230          16637#
CNST0   064370          16689#
CNST1   064434          16726#
CNST2   064432          16725#
COUNT   064455          16746#
CPUERR= 177766           1216#  15170  15336  15366  15414  15589  15618  15626
CR    = 000015            727#  16186  16196  16859  16863
CRLF  = 000200            728#  16157  16196
CSP1    064342          16677#
CSP2    064370          16689#
CST200  064366          16688#
CUA     064447          16740#
DBTA    064630           2767   3313   3325   3363   3375   3563   3571   9626   9646   9665   9676   9689   9700
                         9714   9735   9762   9789   9816   9843   9865  11016  11036  11057  11079  11101  11123
                        11145  11167  11253  16395  16804#
DBTB    064634           3338   3350   3388   3400   3439   3491   3652   3692  16396  16807#
DCS0    064460          16749#
DCS1    064462          16751#
DDISP = 177570            734#   899
DF15    067756           1038  17115#
DF17    067760           1065   1071  17116#
DHEIS1  067222           1194   1201  17046#
DH11    065277           1012  16870#
DH15    067304           1036  17055#
DH17    067336           1063   1069  17060#
DH2     065061            977  16842#
DH23    067373           1018   1024   1030   1075   1110   1117   1138   1173  17065#
DH24    067400           1082   1131   1145   1159   1166  17066#
DH25    067423           1152  17070#
DH26    067437           1096  17073#
DH27    067521           1103  17082#
DH32    067554           1124   1187  17087#
DH4     065072            987  16844#
DH42    067570           1180  17089#
DH44    067607           1043   1050   1057   1089  17092#
DH46    067263           1208  17052#
DISPLA  001042            899#  15983*  16026*
DISPRE  000174            834#
DPTRP = 000001           1219#  15224  15445  15455  15470
DREG    064452          16743#
DSWR  = 177570            733#   898
DTEIS1  067734           1195   1202  17110#
DT1     067764            973  17118#
DT10    067764           1008  17121#
DT11    070024           1013  17129#
DT15    067630           1037  17096#
DT2     067764            978  17119#
DT21    067640           1064  17098#
DT22    067652           1070  17100#
```

```
DT23    067664          1019   1025   1031   1076   1111   1118   1139   1174   17102#
DT24    067670          1044   1051   1058   1083   1090   1132   1146   1160   1167   17103#
DT25    067700          1125   1153   1181   1188   17105#
DT26    067706          1097   17106#
DT27    067722          1104   17108#
DT3     070006           983   17126#
DT4     067764           988   17120#
DT46    067746          1209   17112#
DT5     067772           993   17124#
DT6     067770           998   17123#
DT7     067766          1003   17122#
DWTA    063322          2319   2321   2522   2568   2572   2591   2595   2640   2694   2712   2743   2803   2829
                        2848   2877   2910   3163   3164   3175   3181   3200   3201   3202   3212   3218   3237
                        3238   3239   3249   3255   3274   3275   3286   3292   3515   3527   3537   3549   6679
                        7628   7652   7676   7701   7725   7769   7790   7811   7832   7861   7890   7919   7948
                        7977   8006   8035   8064   8086   8108   8130   8627   8890   8922   8953   8985   9018
                        9055   9092   9129   9166   9203   9244   9281   9318   9350   9382   9414   10502  10529
                       10584  10611  10638  10664  10691  10746  10775  16393  16404#
DWTB    064032          2544   2546   2900   3413   3465   3599   3607   3634   3670   8600   16394  16586#
EA017   002166          1488   1489   1490   1493#
EA020   002242          1522#
EA021   002272          1544#
EA023   002360          1580   1581   1582   1585#
EA032   002564          1690#
EA033   002624          1712#
EA034   002664          1733#
EA041   003030          1814#
EA046   003230          1912#
EME1S1  067065          1193   17029#
EME1S2  067116          1200   17034#
EMTVEC= 000030           822#
EM1     064640           971   16813#
EM10    064724          1006   16825#
EM11    065261          1011   16867#
EM12    065335          1017   16876#
EM13    065374          1023   16882#
EM14    065426          1029   16887#
EM15    065446          1035   16890#
EM16    067035          1042   17024#
EM17    065511          1088   16896#
EM2     064640           976   16814#
EM21    065535          1062   16900#
EM22    065560          1068   16904#
EM23    065607          1074   16908#
EM24    065626          1081   16911#
EM26    065705          1056    1095  16919#
EM27    065732          1102   16923#
EM3     065013           981   16835#
EM30    066002          1109   16930#
EM31    066605          1116   16998#
EM32    066114          1123   16943#
EM33    066145          1130   16948#
EM34    066223          1137   16956#
EM35    066260          1144   16961#
EM36    066346          1151   16970#
EM37    066423          1158   16978#
```

```
EM4     064640    986  16815#
EM40    066516   1165  16988#
EM41    066045   1172  16936#
EM42    066632   1179  17002#
EM43    066720   1186  17011#
EM45    067006   1049  17020#
EM46    067144   1207  17038#
EM5     064666    991  16820#
EM6     064660    996  16819#
EM7     064650   1001  16817#
EOP1    065103  15699  16846#
EOP2    065121  15702  16849#
ERRA    062040   4420  16058#
ERRFLG  063240   4422*  4427*  16058*  16376#
ERRVEC= 000004    815#  14901*  14914*  15948   15949*  15951*  15954*
EX002   001640   1284#
E001    001632   1274#
E003    001656   1294   1295   1296   1299#
E004    001674   1309   1310   1311   1314#
E005    001712   1324   1325   1326   1329#
E006    001724   1343#
E007    001740   1357#
E010    001756   1370   1373#
E011    001776   1386   1389#
E012    002016   1404#
E013    002042   1421#
E014    002064   1436#
E016    002136   1469   1470   1471   1474#
E022    002330   1562   1563   1564   1567#
E024    002414   1606#
E025    002434   1617   1620#
E026    002454   1630   1633#
E027    002474   1647#
E030    002514   1661#
E031    002534   1674#
E035    002714   1753#
E036    002734   1767#
E037    002754   1781#
E040    003002   1798#
E042    003060   1836#
E043    003102   1853#
E044    003130   1869#
E045    003156   1884#
E1015A  002102   1450#
E2002   001644   1288#
E2015   002112   1457#
E2017   002206   1502#
E2020   002252   1528#
E2021   002304   1551#
E2023   002370   1591#
E2032   002574   1696#
E2033   002634   1718#
E2034   002674   1739#
E2041   003040   1820#
E2046   003242   1918#
FAC0.0  064324  16669#
```

```
FAC0.1  064326          16670#
FAC0.2  064330          16671#
FAC0.4  064332          16672#
FAC0.5  064334          16673#
FAC1.0  064236          16640#
FAC1.1  064240          16641#
FAC1.2  064242          16642#
FAC1.3  064244          16643#
FAC1.4  064246          16644#
FAC1.5  064250          16645#
FAC2.0  064272          16656#
FAC2.1  064274          16657#
FAC2.2  064276          16658#
FAC2.3  064300          16659#
FAC2.4  064302          16660#
FAC2.5  064304          16661#
FAC3.0  064202          16626#
FAC3.1  064204          16627#
FAC3.2  064206          16628#
FAC3.3  064210          16629#
FAC3.4  064212          16630#
FAC3.5  064214          16631#
FDST0   064340          16675#
FDST1   064254          16647#
FDST2   064310          16663#
FDST3   064220          16633#
FEA     064336          16674#
FLAG    064450          16741#
FPSHI   064252          16646#
GNS   = ****** U          833   16364   16365   16366   16367
GNWHAM  064226          16636#
HI    = 000200           1222#  15483   15493
HT    = 000011            725#  16155   16196   16819   16820   16825   16835   16842   16844   16849   16870   17046   17052
                        17055   17060   17066   17070   17073   17082   17087   17089   17092
IBUF    063266           4578    4599   16391#
IDENT1  065141           4635   16852#
INCRSV  061202          15834   15849#
INIT    003262           1927#  15715   15848
INSTAP  063636          13855   16523#
IOTVEC= 000020            820#
I002    001642           1280    1286#
I004    001664           1309#
I005    001702           1324#
I006    001720           1339#
I007    001734           1353#
I010    001750           1368#
I011    001766           1383#
I012    002006           1399#
I013    002030           1415#
I014    002060           1432#
I015    002076           1446#
I016    002124           1467#
I017    002154           1486#
I020    002236           1518#
I021    002266           1540#
I022    002312           1560#
```

```
1023    002346          1578#
1024    002404          1602#
1025    002422          1615#
1026    002442          1628#
1027    002466          1643#
1030    002506          1657#
1031    002526          1670#
1032    002560          1686#
1033    002620          1708#
1034    002660          1729#
1035    002710          1749#
1036    002730          1763#
1037    002750          1777#
1040    002772          1793#
1041    003020          1809#
1042    003052          1831#
1043    003074          1848#
1044    003120          1864#
1045    003146          1879#
1046    003212          1903#
JAM     064444         16737#
JMP5    041420         11477   11492   11498#
JMP5A   041466         11507   11524#
LCDTA   064436         16731#
LCUA    064350         16681#
LF    = 000012           726#  16190   16196   16859   16863
LFGIN   064352         16682#
LJAM    064342         16678#
LKCSR = 177546          1261#  12302   12325   12343   12362   12392   12605   12636   12667   12698   12729   12764   12829
                       12876
LO    = 000100          1221#  15483   15493
LPBA    064346         16680#
LSERV   064344         16679#
LTAG    064356         16684#
LWHAM   064354         16683#
MBUF0   063312          2040   2042*   2056   2058*   2117   2135   2155   2174   2191   2196   2208   2213   2231
                        2236   2254    2259   2271    2275   2297   2308   2349   2355*  2368   2373*  2403   2407*
                        2420   2422    2651   2653    2670   2744   2778   2780   2804   2805   2828   2834*  2847
                        2853*  2910*   2911   2930*   2931   2943   2958   2964   2982   2988   3000   3005*  3018
                        3023*  3093    3112   3113    3311   3336   3361   3386   3411   3426   3437   3452   3463
                        3466   3478    3489   3492    3504   3632   3638*  3650   3656*  3668   3674*  3690   3696*
                        4104   4108*   4121*  4125    4482*  4483*  4984   5004   5019   5024   5037   5054   5080
                        5150   5170    5184   5189    5202   5306   5333   5360   5387   5570   5597   5625   5653
                        5681   5709    5737   5749    5767   5797   5799   5810   5828   5830   5854   5856   5867
                        5885   5887    5911   5923    5941   6588   6616   6649   6677   6691   6709   6736   6763
                        6790   6817    6844   6871    6897   6924   6951   6978   7005   7032   7059   7086   7113
                        7140   7167    7194   7221    7378   7380   7387   7405   7407   7427   7435   7453   7474
                        7476   7483    7505   7507    7527   7535   7553   7678*  7684   7727*  7733   7750   7771*
                        7789   7810    7831   7848    7860   7877   7889   7918   7947   7950   7976   7979   8005
                        8034   8063    8066   8085    8088   8107   8129   8260   8289   8322   8351   8513   8542
                        8571   8680    8710   8741    8755   8767   8793   8795   8819   8845   8847   8866   8889
                        8921   8952    8984   9017    9033   9054   9070   9091   9128   9165   9167   9202   9204
                        9243   9280    9317   9319    9349   9351   9381   9413   9443   9464   9485   9506   9508
                        9527   9551    9712   9733    9748   9760   9787   9791   9814   9841   9845   9863   9885
                        9907   9929    9951   9954    9973   9976   10242  10271  10300  10329  10358  10387  10415
                        10444  10473   10744  10773   10802  10832  10862  10896  10926  10956  10986  11034  11055
```

```
                    11077   11099   11102   11121   11143   11146   11165   11187   11208   11211   11230   11251   11254
                    12884*  12906*  12909   13860   13939*  13945   14339   14370   14390   14407*  14442   14443*  14462
                    14479*  14514   14546*  16397   16400#
MBUF1   063316       2154    2173    2711    2716*   2929    3310    3335    3360    3385    7843*   7847    7872*   7876
                     7901*   7905    7930*   7934    7959*   7963    7988*   7992    8017*   8021    8046*   8050    8654
                     8682    8712    9549    10556   10718   10804   10834   10864   10898   10928   10958   10988   12885*
                    12915   12922*  13154   13164   13193   13203   13861   13905*  13911   16398   16402#
MD      064440      16733#
MED   = 076600       1212#  13798   14905   14918   14949   14954   14957   14961   14995   15008   15037   15040   15048
                    15082   15085   15087   15090   15101   15114   15117   15121   15133   15136   15138   15159   15168
                    15183   15190   15202   15229   15231   15233   15238   15244   15250   15256   15262   15295   15300
                    15334   15350   15364   15381   15412   15453   15459   15468   15480   15501   15512   15526   15534
                    15540   15571   15587   15609   15624   15635   15653   15668   15748
MEDHLT  055632      14920   14921   14922   14924#
MEDT1   055642      14941#
MEDT10  056206      15042#
MEDT11  056260      15078#  15140
MEDT3   056042      14992#
MEDT4   056100      15006#
MEDO    055612      14917#
MED1    055514      14898#
MEMERR= 177744       1215#  15303   15307   15492
NOPS    064404      16704#
NUA     064456      16747#
OBUF    063256       4579    4600   16386#
ONCE    063254       1924*   1929    4631    4633*  16383#
ORTAB   063474      14184   14189   16465#
PABORT= 000200       1220#  15445
PA1500  064162      16601#
PA1716  064160      16600#
PBA     064446      16739#
PIRQ  = 177772        732#
PIRQVE= 000240        826#
PRIFLG  063236       1927    4398*   4407*  16113*  16375#
PRINA   062202       4400   16113#
PR0   = 000000        749#
PR1   = 000040        750#
PR2   = 000100        751#
PR3   = 000140        752#
PR4   = 000200        753#
PR5   = 000240        754#
PR6   = 000300        755#
PR7   = 000340        756#
PS    = 177776        729#    730
PSW   = 177776        730#   1429    1443    1464    1482    1493    1502    1511    1560*   1574    1598    1602    1615
                     1628    1681    1682*   1703    1704*   1724    1725*   1788    1804    1994*   2009*   2023    4223*
                     4231    4235*   4254*   4259    4263*   4277*   4285    4289*   4296*   4303*   4338*   4346    4350*
                     4357*   4364*   4377*   4382    4385*   4636*   4678    4684    4698    4706    4720    4726    4740
                     4748   12107   12114*  12122   12144   12151*  12159   12208   12211   12268   12271   12368*  12399*
                    12414*  12425*  12431   12443   12446*  12458*  12464   12476   12479*  12504*  12510*  12527   12534*
                    12561*  12567*  12584   12591*  12610*  12625*  12641*  12656*  12672*  12687*  12703*  12718*  12734*
                    12749*  12769*  12785*  12801*  12817*  12837*  12847*  12883*  12904*  12940*  14679*  14708   14709
                    14899*  14917*  14941*  15772*
PSWHOL  064050      16595#
PTRP1   057712      15456   15467#
PWRUP   061054      15762   15771#
```

```
PWRVEC= 000024            821#  15726*  15727*  15736*  15742*  15757*  15758*
RCSR  = 177560           1257#   4489    4575
RDBR  = 177562           1258#
RDFLAG= 000144           1233#  15038   15088   15115
RDLCUA= 000103           1241#  15102
RDLDAT= 000106           1247#  15257   15527
RDLFGI= 000104           1243#  15382   15541   15654   15669
RDLJAM= 000100           1235#  15169   15335   15365   15413   15588   15625
RDLPBA= 000102           1239#  15184   15251   15351   15502
RDLSER= 000101           1237#  15191   15481
RDLTAG= 000107           1249#  15263   15513
RDLWHA= 000105           1245#
RDWHAM= 000022           1231#  15083   15134
RES     064457          16748#
RESTAR  061176          15837   15846   15848#  15851   15905   15912
RESVEC= 000010            816#  14902*  14915*
REV     064453          16744#
RSBERT  061150          15840#  15908
RSERR   061122           4453   13835   14915   15833#
RSMSG   065233          15839   16863#
RSVFLG  063246           4441*   4447*  15830*  16379#
RSVTST  061114           4438   15830#
RT1A    064372          16690#
RT1B    064376          16692#
RT2A    064374          16691#
RT2B    064400          16693#
RVECT   064322          16668#
RZERO   064320          16667#
R0A     064422          16721#
R0B     064426          16723#
R1A     064166          16620#
R1B     064256          16650#
R2A     064170          16621#
R2B     064260          16651#
R3A     064172          16622#
R3B     064262          16652#
R4A     064174          16623#
R4B     064264          16653#
R5A     064176          16624#
R5B     064266          16654#
R6A     064200          16625#
R6B     064270          16655#
R7A     064424          15006   16722#
R7B     064430          15018   16724#
SCOFLG  063244           4314*   4321*  15992*  16378#
SCOPEA  061612           4315   15992#
SELTST  063242          15938*  15939*  15940   16377#
SERV    064445          16738#
SOBERR  043012          11972   11981#
SOB1    042764          11967#  12050
SOB2    043000          11962   11965   11975#
SOB3    043156          11967   12048#
SOB4    043160          12046   12050#
SOB5    042772          11971#  11975
SREG    064454          16745#
STACK = 001000            720#   1428    1915    1931   14900   15771
```

```
START   001630            837    1271#  15773
STGBLK  064056          16598#
STKLMT= 177774            731#
SWB01 = 000071           1251#  15078   15096   15100
SWR     001040            898#   4628*   4629   15734   15750*  15836   15845   15904   15936   15938   15943   15961   15968
                        16031   16043   16046
SWREG   000176            835#
SW0   = 000001            784#
SW00  = 000001            774#    784
SW01  = 000002            773#    783
SW02  = 000004            772#    782
SW03  = 000010            771#    781
SW04  = 000020            770#    780
SW05  = 000040            769#    779
SW06  = 000100            768#    778
SW07  = 000200            767#    777
SW08  = 000400            766#    776
SW09  = 001000            765#    775
SW1   = 000002            783#
SW10  = 002000            764#  15936
SW11  = 004000            763#
SW12  = 010000            762#   4629   15836   15904
SW13  = 020000            761#
SW14  = 040000            760#
SW15  = 100000            759#
SW2   = 000004            782#
SW3   = 000010            781#
SW4   = 000020            780#
SW5   = 000040            779#
SW6   = 000100            778#
SW7   = 000200            777#
SW8   = 000400            776#
SW9   = 001000            775#
TABBEG  064052          16596#
TABEND  064054          16597#
TAG   = 000040           1223#  15483   15493
TBITVE= 000014            817#
TBL2    064166          14942   16617#
TBL3    064404          14992   16703#
TBL4    064422          16720#
TBL5    064436          15107   16729#
TBL6    064464          15108   16757#
TBL7    064536          15044   16785#
TBSER   061070          13843   15779#
TKVEC = 000060            824#
TLOC1   064046          15220   15235   15286   15293   15447   15457   15503   15507   15515   16594#
TLOC2   064164          16602#
TPVEC = 000064            825#
TRAPVE= 000034            823#
TRTVEC= 000014            818#
TSET    052422          13843#
TST0    003316           1939#   4637
TST1    003332           1944    1952#
TST10   003532           2047    2054#
TST100  007200           3225    3234#
TST101  007274           3262    3271#
```

TST102	007366	3299	3308#
TST103	007446	3326	3333#
TST104	007526	3351	3358#
TST105	007606	3376	3383#
TST106	007666	3401	3409#
TST107	007744	3427	3435#
TST11	003564	2063	2071#
TST110	010022	3453	3461#
TST111	010102	3479	3487#
TST112	010162	3505	3513#
TST113	010224	3528	3535#
TST114	010266	3550	3558#
TST115	010356	3594#	
TST116	010446	3630#	
TST117	010510	3641	3648#
TST12	003622	2085	2093#
TST120	010552	3659	3666#
TST121	010614	3677	3684#
TST122	010670	3699	3707#
TST123	010714	3715	3723#
TST124	010752	3743	3751#
TST125	010770	3757	3765#
TST126	011004	3771	3779#
TST127	011020	3785	3793#
TST13	003656	2107	2115#
TST130	011040	3800	3808#
TST131	011060	3815	3823#
TST132	011076	3830	3838#
TST133	011114	3844	3852#
TST134	011132	3859	3867#
TST135	011150	3874	3882#
TST136	011170	3889	3897#
TST137	011210	3904	3912#
TST14	003710	2125	2133#
TST140	011230	3919	3927#
TST141	011250	3934	3942#
TST142	011264	3948	3956#
TST143	011304	3963	3971#
TST144	011324	3978	3986#
TST145	011344	3993	4001#
TST146	011362	4008	4016#
TST147	011376	4022	4030#
TST15	003744	2143	2151#
TST150	011416	4037	4045#
TST151	011436	4052	4060#
TST152	011456	4067	4075#
TST153	011524	4093	4101#
TST154	011566	4111	4119#
TST155	011622	4128	4136#
TST156	011710	4164	4172#
TST157	011744	4185	4193#
TST16	004010	2162	2170#
TST160	012012	4209	4217#
TST161	012100	4240	4248#
TST162	012152	4261	4270#
TST163	012274	4301	4310#

TST164	012342	4331#	
TST165	012464	4362	4371#
TST166	012540	4394#	
TST167	012620	4408	4416#
TST17	004054	2181	2189#
TST170	012664	4428	4435#
TST171	012746	4459#	
TST172	013016	4472	4480#
TST173	013116	4508#	
TST174	013156	4518	4529#
TST175	013210	4538	4547#
TST176	013244	4556	4573#
TST177	013556	4645#	
TST2	003350	1958	1966#
TST20	004110	2199	2206#
TST200	013600	4652	4659#
TST201	013620	4666	4673#
TST202	013660	4686	4693#
TST203	013722	4708	4715#
TST204	013760	4728	4735#
TST205	014020	4750	4757#
TST206	014042	4764	4771#
TST207	014064	4779	4786#
TST21	004154	2221	2229#
TST210	014106	4794	4801#
TST211	014130	4809	4816#
TST212	014154	4824	4831#
TST213	014176	4838	4845#
TST214	014220	4853	4860#
TST215	014242	4868	4875#
TST216	014264	4883	4890#
TST217	014332	4909	4916#
TST22	004214	2244	2252#
TST220	014374	4930	4937#
TST221	014442	4956	4963#
TST222	014474	4973	4980#
TST223	014622	5025	5032#
TST224	014656	5042	5049#
TST225	014726	5068	5075#
TST226	014764	5086	5093#
TST227	015034	5112	5119#
TST23	004250	2261	2269#
TST230	015104	5138	5145#
TST231	015234	5190	5197#
TST232	015310	5217	5224#
TST233	015354	5243	5250#
TST234	015424	5269	5276#
TST235	015472	5295	5302#
TST236	015542	5322	5329#
TST237	015616	5349	5356#
TST24	004310	2283	2291#
TST240	015672	5376	5383#
TST241	015744	5403	5410#
TST242	016014	5429	5436#
TST243	016062	5455	5462#
TST244	016132	5481	5488#

TST245	016200	5507	5514#
TST246	016250	5533	5540#
TST247	016320	5559	5566#
TST25	004360	2309	2317#
TST250	016376	5586	5593#
TST251	016452	5613	5621#
TST252	016530	5641	5649#
TST253	016604	5669	5677#
TST254	016662	5697	5705#
TST255	016740	5725	5733#
TST256	017020	5755	5763#
TST257	017072	5781	5789#
TST26	004434	2338	2347#
TST260	017170	5816	5824#
TST261	017244	5842	5850#
TST262	017330	5873	5881#
TST263	017404	5899	5907#
TST264	017464	5929	5937#
TST265	017534	5954	5962#
TST266	017600	5981	5988#
TST267	017646	6008	6015#
TST27	004470	2358	2366#
TST270	017714	6034	6041#
TST271	017760	6059	6066#
TST272	020030	6085	6092#
TST273	020076	6111	6118#
TST274	020144	6137	6144#
TST275	020214	6163	6170#
TST276	020262	6189	6196#
TST277	020332	6215	6222#
TST3	003364	1971	1978#
TST30	004526	2376	2384#
TST300	020376	6240	6247#
TST301	020444	6266	6273#
TST302	020514	6292	6299#
TST303	020560	6318	6325#
TST304	020630	6344	6351#
TST305	020700	6369	6376#
TST306	020744	6395	6402#
TST307	021014	6421	6428#
TST31	004552	2392	2400#
TST310	021062	6447	6454#
TST311	021132	6473	6480#
TST312	021200	6499	6506#
TST313	021250	6525	6532#
TST314	021320	6551	6558#
TST315	021366	6577	6584#
TST316	021440	6604	6612#
TST317	021516	6633	6641#
TST32	004606	2410	2418#
TST320	021604	6665	6673#
TST321	021666	6697	6705#
TST322	021744	6724	6732#
TST323	022020	6751	6759#
TST324	022074	6778	6786#
TST325	022152	6805	6813#

TST326	022226	6832	6840#
TST327	022304	6859	6867#
TST33	004646	2429	2437#
TST330	022356	6885	6893#
TST331	022432	6912	6920#
TST332	022510	6939	6947#
TST333	022562	6966	6974#
TST334	022640	6993	7001#
TST335	022716	7020	7028#
TST336	022770	7047	7055#
TST337	023046	7074	7082#
TST34	004702	2450	2458#
TST340	023122	7101	7109#
TST341	023200	7128	7136#
TST342	023254	7155	7163#
TST343	023332	7182	7190#
TST344	023410	7209	7217#
TST345	023464	7236	7244#
TST346	023534	7263	7270#
TST347	023604	7289	7296#
TST35	004736	2471	2478#
TST350	023654	7315	7322#
TST351	023724	7341	7348#
TST352	023774	7367	7374#
TST353	024054	7393	7401#
TST354	024124	7415	7423#
TST355	024200	7441	7449#
TST356	024244	7462	7470#
TST357	024324	7489	7497#
TST36	004772	2491	2499#
TST360	024406	7515	7523#
TST361	024462	7541	7549#
TST362	024526	7562	7570#
TST363	024600	7590	7597#
TST364	024654	7617	7624#
TST365	024722	7641	7648#
TST366	024770	7665	7672#
TST367	025044	7690	7697#
TST37	005026	2512	2520#
TST370	025112	7714	7721#
TST371	025166	7739	7746#
TST372	025226	7758	7765#
TST373	025272	7778	7785#
TST374	025336	7798	7806#
TST375	025402	7819	7827#
TST376	025470	7849	7856#
TST377	025556	7878	7885#
TST4	003402	1984	1992#
TST40	005066	2534	2542#
TST400	025646	7907	7914#
TST401	025736	7936	7943#
TST402	026024	7965	7972#
TST403	026112	7994	8001#
TST404	026202	8023	8030#
TST405	026272	8052	8059#
TST406	026344	8073	8081#

TST407	026416	8095	8103#
TST41	005132	2557	2566#
TST410	026466	8117	8125#
TST411	026536	8139	8147#
TST412	026602	8166	8173#
TST413	026652	8193	8200#
TST414	026726	8220	8227#
TST415	027002	8247	8254#
TST416	027056	8275	8283#
TST417	027136	8304	8312#
TST42	005174	2580	2589#
TST420	027232	8337	8345#
TST421	027314	8366	8374#
TST422	027364	8394	8401#
TST423	027434	8421	8428#
TST424	027510	8448	8455#
TST425	027564	8475	8482#
TST426	027640	8502	8509#
TST427	027720	8530	8538#
TST43	005236	2603	2612#
TST430	030000	8559	8567#
TST431	030060	8587	8595#
TST432	030132	8615	8622#
TST433	030202	8642	8649#
TST434	030260	8669	8676#
TST435	030344	8698	8706#
TST436	030432	8729	8737#
TST437	030506	8756	8763#
TST44	005274	2633#	
TST440	030564	8782	8789#
TST441	030640	8808	8815#
TST442	030716	8834	8841#
TST443	030766	8854	8862#
TST444	031036	8875	8883#
TST445	031122	8906	8915#
TST446	031206	8938	8947#
TST447	031272	8970	8979#
TST45	005322	2641	2649#
TST450	031356	9002	9011#
TST451	031452	9039	9048#
TST452	031546	9076	9085#
TST453	031644	9113	9122#
TST454	031742	9150	9159#
TST455	032036	9187	9196#
TST456	032132	9224	9233#
TST457	032242	9265	9274#
TST46	005364	2660	2668#
TST460	032340	9302	9311#
TST461	032430	9334	9343#
TST462	032520	9366	9375#
TST463	032610	9398	9407#
TST464	032700	9430	9439#
TST465	032740	9452	9460#
TST466	033000	9473	9481#
TST467	033042	9494	9502#
TST47	005420	2678	2686#

TST470	033104	9515	9523#
TST471	033146	9536	9545#
TST472	033212	9558	9566#
TST473	033264	9586	9593#
TST474	033336	9614	9621#
TST475	033374	9634	9641#
TST476	033434	9653	9660#
TST477	033502	9677	9684#
TST5	003424	1999	2007#
TST50	005462	2701	2709#
TST500	033550	9701	9708#
TST501	033614	9721	9729#
TST502	033672	9749	9756#
TST503	033752	9775	9783#
TST504	034030	9802	9810#
TST505	034110	9829	9837#
TST506	034162	9851	9859#
TST507	034234	9873	9881#
TST51	005520	2718	2726#
TST510	034302	9895	9903#
TST511	034350	9917	9925#
TST512	034420	9939	9947#
TST513	034470	9961	9969#
TST514	034542	9983	9991#
TST515	034616	10011	10018#
TST516	034676	10042	10049#
TST517	034752	10069	10076#
TST52	005544	2741#	
TST520	035020	10096	10103#
TST521	035076	10123	10131#
TST522	035150	10151	10158#
TST523	035222	10178	10185#
TST524	035274	10205	10212#
TST525	035346	10231	10238#
TST526	035430	10259	10267#
TST527	035504	10288	10296#
TST53	005600	2751	2759#
TST530	035566	10317	10325#
TST531	035642	10346	10354#
TST532	035724	10375	10383#
TST533	036006	10404	10411#
TST534	036070	10432	10440#
TST535	036150	10461	10469#
TST536	036230	10489	10497#
TST537	036304	10517	10524#
TST54	005632	2768	2776#
TST540	036354	10544	10551#
TST541	036434	10572	10579#
TST542	036504	10599	10606#
TST543	036556	10626	10633#
TST544	036630	10653	10659#
TST545	036702	10679	10686#
TST546	036754	10706	10713#
TST547	037032	10733	10740#
TST55	005700	2793	2801#
TST550	037114	10761	10769#

TST551	037172	10790	10798#			
TST552	037260	10820	10828#			
TST553	037340	10850	10858#			
TST554	037426	10880	10888#			
TST555	037526	10914	10922#			
TST556	037612	10944	10952#			
TST557	037676	10974	10982#			
TST56	005750	2818	2826#			
TST560	037762	11003	11011#			
TST561	040020	11023	11030#			
TST562	040064	11043	11051#			
TST563	040132	11065	11073#			
TST564	040202	11087	11095#			
TST565	040252	11109	11117#			
TST566	040322	11131	11139#			
TST567	040374	11153	11161#			
TST57	006010	2837	2845#			
TST570	040446	11175	11183#			
TST571	040510	11196	11204#			
TST572	040560	11218	11226#			
TST573	040622	11239	11247#			
TST574	040672	11261	11269#			
TST575	040732	11279	11284	11291#		
TST576	040772	11301	11306	11313#		
TST577	041042	11323	11333	11340#		
TST6	003444	2013	2021#			
TST60	006062	2862	2869#			
TST600	041102	11350	11355	11362#		
TST601	041160	11372	11382	11385	11394#	
TST602	041226	11404	11409	11412	11420#	
TST603	041304	11430	11434	11444	11451#	
TST604	041344	11461	11466	11473#		
TST605	041422	11483	11493	11496	11503#	
TST606	041470	11513	11518	11521	11529#	
TST607	041536	11539	11544	11547	11555#	
TST61	006124	2884	2892#			
TST610	041604	11565	11570	11573	11581#	
TST611	041660	11591	11594	11599	11602	11611#
TST612	041734	11621	11624	11629	11632	11641#
TST613	042012	11657	11669#			
TST614	042100	11700#				
TST615	042162	11729#				
TST616	042232	11754#				
TST617	042310	11770	11783#			
TST62	006156	2901	2908#			
TST620	042374	11799	11816#			
TST621	042456	11835	11848#			
TST622	042542	11864	11881#			
TST623	042622	11897	11909#			
TST624	042710	11926	11943#			
TST625	042740	11951	11959#			
TST626	043014	11977	11980	11986#		
TST627	043050	11998	12005#			
TST63	006216	2918	2926#			
TST630	043104	12017	12024#			
TST631	043140	12036	12043#			

```
TST632  043174          12055   12062#
TST633  043276          12092   12103#
TST634  043402          12129   12140#
TST635  043510          12166   12177#
TST636  043666          12193   12201   12213   12222   12237#
TST637  044044          12253   12261   12273   12282   12297#
TST64   006274           2944    2952#
TST640  044126          12321#
TST641  044162          12331   12339#
TST642  044220          12349   12357#
TST643  044332          12387#
TST644  044450          12419#
TST645  044566          12452#
TST646  044702          12485#
TST647  045106          12542#
TST65   006352           2972    2980#
TST650  045310          12599#
TST651  045426          12630#
TST652  045544          12661#
TST653  045662          12692#
TST654  046000          12723#
TST655  046116          12754#
TST656  046250          12790#
TST657  046370          12825#
TST66   006406           2990    2998#
TST660  046530          12870#
TST661  047030          12953#
TST662  047076          12973#
TST663  047232          13013#
TST664  047346          13051#
TST665  047430          13076#
TST666  047512          13101#
TST667  047574          13126#
TST67   006444           3008    3016#
TST670  047660          13150#
TST671  047766          13181   13189#
TST672  050074          13219   13227#
TST673  050234          13258   13279#
TST674  050416          13327   13339#
TST675  050600          13387   13399#
TST676  050674          13429#
TST677  051000          13461#
TST7    003476           2031    2038#
TST70   006504           3026    3034#
TST700  051104          13493#
TST701  051164          13516#
TST702  051244          13539#
TST703  051324          13561#
TST704  051424          13594#
TST705  051530          13612   13629#
TST706  051636          13647   13664#
TST707  051744          13682   13699#
TST71   006534           3043    3051#
TST710  052032          13725#
TST711  052120          13752#
TST712  052206          13779#
```

```
TST713  052304          13809#
TST714  052436          13852#
TST715  052526          13879#
TST716  052556          13894#
TST717  052644          13919#
TST72   006604           3075#
TST720  052700          13935#
TST721  052742          13952#
TST722  053010          14001#
TST723  053074          14061#
TST724  053160          14121#
TST725  053244          14181#
TST726  053330          14227#
TST727  053404          14245   14271#
TST73   006634           3083    3091#
TST730  053460          14289   14301#
TST731  053544          14324   14331#
TST732  053634          14355   14362#
TST733  053732          14391   14398#
TST734  054036          14427   14434#
TST735  054142          14463   14470#
TST736  054246          14499   14506#
TST737  054340          14530   14537#
TST74   006672           3101    3109#
TST740  054422          14554   14561#
TST741  054506          14583   14590#
TST742  054574          14612   14619#
TST743  054656          14641   14648#
TST744  054730          14663   14674#
TST745  055046          14706   14716#
TST746  055116          14734   14740#
TST747  055166          14759   14765#
TST75   006734           3120    3128#
TST750  055236          14783   14789#
TST751  055324          14811   14817#
TST752  055406          14840   14846#
TST753  055506          14872   14895#
TST754  055634          14938#
TST755  056034          14989#
TST756  056164          15033#
TST757  056252          15075#
TST76   007012           3152    3160#
TST760  056554          15153#
TST761  056732          15216#
TST762  057142          15283#
TST763  057274          15324#
TST764  057502          15400#
TST765  057600          15441#
TST766  060160          15566#
TST767  060402          15641#
TST77   007104           3188    3197#
TYPE   = 104401           4403    4634   15699  15702  15705  15759  15838  15842  15906  16034  16069  16086  16088
                         16091   16093  16097  16104  16160  16257  16364#
TYPOC  = 104402          15701   15704  15841  16077  16101  16365#
TYPON  = 104404          16367#
TYPOS  = 104403          16366#
```

```
UBREAK= 177770           1214#   1932*  14898*  15078*  15112*  15142*
UCNSCT  064474          16763#
UCOUNT  064522          16774#
UCUA    064506          16768#
UDCS0   064530          16777#
UDCS1   064534          16779#
UDREG   064514          16771#
UFLAG   064510          16769#
UINIT   064532          16778#
UJAM    064500          16765#
ULCDTA  064464          16759#
UM    = 140000           1213#
UMD     064470          16761#
UNUA    064524          16775#
UPBA    064504          16767#
URES    064526          16776#
UREV    064516          16772#
UR6A    064216          16632#
UR6B    064306          16662#
USERV   064502          16766#
USREG   064520          16773#
VADR    064156          16599#
WCNSSW= 000226           1230#  15749
WCSADR  064316          16666#
WCSA.0  064222          16634#
WCSA.1  064224          16635#
WCSB.0  064312          16664#
WCSB.1  064314          16665#
WRFLAG= 000344           1234#  15041   15091   15118   15137
WRLCUA= 000303           1242#
WRLDAT= 000306           1248#  15232
WRLFG1= 000304           1244#
WRLJAM= 000300           1236#
WRLPBA= 000302           1240#  15230
WRLSER= 000301           1238#
WRLTAG= 000307           1250#  15234
WRLWHA= 000305           1246#
WRWHAM= 000222           1232#  15086   15139   15160   15203   15454   15535   15572   15610   15636
WWP   = 000100           1218#  15222   15288   15290   15449   15451
XCSR  = 177564           1259#   4510    4531    4549   12424   12447*  12457   12480*  12491   12548   12796   12877
XDBR  = 177566           1260#
$APTHD  000700            848     854#
$ASTAT= ****** U        16309   16324
$ATYC   062744          16280   16282#
$ATY1   062720          16278#
$ATY3   062726          16145   16279#
$ATY4   062736          16039   16281#
$AUTOB  001034            895#
$BDADR  001022            890#
$BDDAT  001026            892#
$CHARC  062466          16162*  16172*  16179   16188*  16193#
$CKSWR= ****** U        16370
$CMTAG  001000            878#
$CM1  = 000006            910#    911#    912#    913#    914#    915#    916#
$CM2  = 000014            910#    911#    912#    913#    914#    915#    916#
$CM3  = 000006            908#    910
```

```
$CM4   = 000005            916#    917#    918#    919#    920#    921#
$CPUOP  001146             945#
$CRLF   001115             924#  15705   15843   16034   16057   16069   16088   16093   16097   16161   16196
$DEVCT  001130             936#
$DOAGN  060654           15695   15707   15713#
$ENDAD  060644             867   15709#
$ENDCT  060600           15697#
$ENULL  060660           15716#
$ENV    001140             941#  16036   16140   16287   16311
$ENVM   001141             942#   4626   16142   16147   16289
$EOP    060550           15688#
$EOPCT  060572           15694#  15698
$ERFLG  001003             881#  15921   15957   15959   15965*  15991   16024*  16057
$ERMAX  001015             887#  15959   15982*  15991
$ERROR  061620            4620   15648   16011#
$ERRPC  001016             888#  16028*  16029*  16030   16057   16075   17096   17098   17100   17102   17103   17105   17106
                         17108   17112   17124   17126   17129
$ERRTB  001150             968#  16083
$ERRTY  062046           16033   16068#
$ERTTL  001012             885#   1925*  15703   15844*  16027*  16057
$ESCAP  001112             922#  15932*  15934*  15981*  16049   16051   16057
$ETABL  001140             940#
$ETEND  001150             860     952#
$FATAL  001122             933#  16315*
$FFLG   063164           16278*  16281*  16309   16318*  16326#
$FILLC  001056             906#  16165   16196
$FILLS  001055             905#  16196
$GDADR  001020             889#
$GDDAT  001024             891#
$GET42  060634           15706#
$GTSWR= ****** U         16369
$HD    = 000000            705
$HIBTS  000700             855#
$ICNT   001004             882#  15972*  15973   15975*  15990
$ILLUP  061036           15726   15742   15764#
$INTAG  001035             896#
$ITEMB  001014             886#  16030*  16038   16057   16072
$LF     001116             925#  16057   16196
$LFLG   063163           16319*  16325#
$LPADR  001006             883#   4637*  15963*  15979*  15984   15990
$LPERR  001010             884#  11646*  11674*  11709*  11734*  11759*  11788*  11821*  11853*  11886*  11914*  12067*  12109*
                         12146*  12303*  12364*  12393*  12490*  12547*  12604*  12635*  12666*  12697*  12728*  12763*  12795*
                         12831*  12875*  12958*  13131*  13498*  13521*  13544*  13566*  13706*  13721*  13732*  13747*  13759*
                         13774*  13786*  13804*  13816*  13834*  13858*  13874*  13903*  13957*  14005*  14027*  14065*  14087*
                         14125*  14147*  14185*  14207*  14233*  14252*  14277*  14296*  15109*  15140*  15963   15980*  15990
                         16048
$MAIL   001120             856     860     931#  15978   16036   16140
$MBADR  000702             856#
$MFLG   063162           16279*  16285   16320*  16324#
$MSGAD  001134             938#  16295*  16298
$MSGLG  001136             939#  16300*
$MSGTY  001120             932#  16293   16301*  16313   16317*
$MXCNT  061610           15976   15990#
$NULL   001054             904#  16167   16196
$NWTST= 000001            1936#   1949#   1963#   1975#   1989#   2004#   2018#   2035#   2051#   2068#   2090#   2112#   2130#
                          2148#   2167#   2186#   2203#   2226#   2249#   2266#   2288#   2314#   2344#   2363#   2381#   2397#
```

```
                2415#  2434#  2455#  2475#  2496#  2517#  2539#  2563#  2586#  2609#  2630#  2646#  2665#
                2683#  2706#  2723#  2738#  2756#  2773#  2798#  2823#  2842#  2866#  2889#  2905#  2923#
                2949#  2977#  2995#  3013#  3031#  3048#  3072#  3088#  3106#  3125#  3157#  3194#  3231#
                3268#  3305#  3330#  3355#  3380#  3406#  3432#  3458#  3484#  3510#  3532#  3555#  3591#
                3627#  3645#  3663#  3681#  3704#  3720#  3748#  3762#  3776#  3790#  3805#  3820#  3835#
                3849#  3864#  3879#  3894#  3909#  3924#  3939#  3953#  3968#  3983#  3998#  4013#  4027#
                4042#  4057#  4072#  4098#  4116#  4133#  4169#  4190#  4214#  4245#  4267#  4307#  4328#
                4368#  4391#  4413#  4432#  4456#  4477#  4505#  4526#  4544#  4562#  4564    4642#  4656#
                4670#  4690#  4712#  4732#  4754#  4768#  4783#  4798#  4813#  4828#  4842#  4857#  4872#
                4887#  4913#  4934#  4960#  4977#  5029#  5046#  5072#  5090#  5116#  5142#  5194#  5221#
                5247#  5273#  5299#  5326#  5353#  5380#  5407#  5433#  5459#  5485#  5511#  5537#  5563#
                5590#  5618#  5646#  5674#  5702#  5730#  5760#  5786#  5821#  5847#  5878#  5904#  5934#
                5959#  5985#  6012#  6038#  6063#  6089#  6115#  6141#  6167#  6193#  6219#  6244#  6270#
                6296#  6322#  6348#  6373#  6399#  6425#  6451#  6477#  6503#  6529#  6555#  6581#  6609#
                6638#  6670#  6702#  6729#  6756#  6783#  6810#  6837#  6864#  6890#  6917#  6944#  6971#
                6998#  7025#  7052#  7079#  7106#  7133#  7160#  7187#  7214#  7241#  7267#  7293#  7319#
                7345#  7371#  7398#  7420#  7446#  7467#  7494#  7520#  7546#  7567#  7594#  7621#  7645#
                7669#  7694#  7718#  7743#  7762#  7782#  7803#  7824#  7853#  7882#  7911#  7940#  7969#
                7998#  8027#  8056#  8078#  8100#  8122#  8144#  8170#  8197#  8224#  8251#  8280#  8309#
                8342#  8371#  8398#  8425#  8452#  8479#  8506#  8535#  8564#  8592#  8619#  8646#  8673#
                8703#  8734#  8760#  8786#  8812#  8838#  8859#  8880#  8912#  8944#  8976#  9008#  9045#
                9082#  9119#  9156#  9193#  9230#  9271#  9308#  9340#  9372#  9404#  9436#  9457#  9478#
                9499#  9520#  9542#  9563#  9590#  9618#  9638#  9657#  9681#  9705#  9726#  9753#  9780#
                9807#  9834#  9856#  9878#  9900#  9922#  9944#  9966#  9988#  10015#  10046#  10073#  10100#
               10128#  10155#  10182#  10209#  10235#  10264#  10293#  10322#  10351#  10380#  10408#  10437#  10466#
               10494#  10521#  10548#  10576#  10603#  10630#  10656#  10683#  10710#  10737#  10766#  10795#  10825#
               10855#  10885#  10919#  10949#  10979#  11008#  11027#  11048#  11070#  11092#  11114#  11136#  11158#
               11180#  11201#  11223#  11244#  11266#  11288#  11310#  11337#  11359#  11391#  11417#  11448#  11470#
               11500#  11526#  11552#  11578#  11608#  11638#  11666#  11697#  11726#  11751#  11780#  11813#  11845#
               11878#  11906#  11940#  11956#  11983#  12002#  12021#  12040#  12059#  12100#  12137#  12174#  12234#
               12294#  12318#  12336#  12354#  12384#  12416#  12449#  12482#  12539#  12596#  12627#  12658#  12689#
               12720#  12751#  12787#  12819#  12821   12863#  12865   12950#  12970#  13010#  13048#  13073#  13098#
               13123#  13147#  13186#  13224#  13276#  13336#  13396#  13426#  13458#  13490#  13513#  13536#  13558#
               13589#  13591   13626#  13661#  13696#  13722#  13749#  13776#  13806#  13846#  13848   13876#  13891#
               13916#  13932#  13949#  13969#  13971   14029#  14031   14089#  14091   14149#  14151   14209#  14211
               14254#  14256   14298#  14328#  14359#  14395#  14431#  14467#  14503#  14534#  14558#  14587#  14616#
               14645#  14667#  14669   14713#  14737#  14762#  14786#  14814#  14843#  14888#  14890   14926#  14928
               14977#  14979   15023#  15025   15058#  15060   15145#  15147   15206#  15208   15276#  15278   15313#
               15315   15390#  15392   15429#  15431   15551#  15553   15638#
$OCNT   062714        16229*  16258*  16271#
$OMODE  062716        16224*  16228*  16233   16236*  16247*  16273#
$OVER   061560        15941   15944   15964   15974   15983#
$PASS   001126          935#   1926*  15691*  15692*  15700   15716   15970   15991
$PASTM  000706          858#
$POWER  061044        15760   15767#
$PWRAD  061032        15762#
$PWRDN  060664         4624   15726#  15757
$PWRMG  061026        15760#
$PWRUP  060736        15736   15742#
$QUES   001114          923#  16057   16196
$RDCHR= ****** U      16370
$RDDEC= ****** U      16370
$RDLIN= ****** U      16370
$RDOCT= ****** U      16370
$REGAD  001060          908#  16013   17110   17124   17126
$REGO   001062          910#  15446*  15457*  15474   15544*  17098   17103   17105   17106   17112   17124   17126   17129
```

```
$REG1   001064            911#  17103  17106  17108  17110  17124  17126
$REG2   001066            912#  17106  17108  17110  17123
$REG3   001070            913#  17106  17108  17110  17122  17126
$REG4   001072            914#  17110  17121  17126
$REG5   001074            915#   1933* 12189* 12197* 12209* 12227* 12249* 12257* 12269* 12287* 12988* 12996* 13003*
                        13027* 13035* 13042* 13066* 13091* 13116* 13292* 13302* 13312* 13322* 13352* 13362* 13372*
                        13382* 13410* 13416* 13420* 13442* 13451* 13474* 13483* 16053* 17124
$RTNAD  060656          15715#
$R2A  = ****** U        16370
$SAVRE= ****** U        16370
$SAVR6  061042          15735* 15743  15744* 15745* 15766#
$SCOPE  061260           4618  15663  15929#
$SETUP= 000000          15690  15930  16012  16046  16053
$SVLAD  061524          15952  15977#
$SVPC = 000714            865#   870
$SWR  = 165000            695#   705    709    710    711    712    713    714    715    716    921    922    923
                         1941   1954   1968   1980   1994   2009   2023   2040   2056   2073   2095   2117   2135
                         2153   2172   2191   2208   2231   2254   2271   2293   2319   2349   2368   2386   2402
                         2420   2439   2460   2480   2501   2522   2544   2568   2591   2614   2635   2651   2670
                         2688   2711   2728   2743   2761   2778   2803   2828   2847   2871   2894   2910   2928
                         2954   2982   3000   3018   3036   3053   3077   3093   3111   3130   3162   3199   3236
                         3273   3310   3335   3360   3385   3411   3437   3463   3489   3515   3537   3560   3596
                         3632   3650   3668   3686   3709   3725   3753   3767   3781   3795   3810   3825   3840
                         3854   3869   3884   3899   3914   3929   3944   3958   3973   3988   4003   4018   4032
                         4047   4062   4077   4103   4121   4138   4174   4195   4219   4250   4272   4312   4333
                         4373   4396   4418   4437   4461   4482   4510   4531   4549   4575   4649   4663   4677
                         4697   4719   4739   4761   4775   4790   4805   4820   4835   4849   4864   4879   4894
                         4920   4941   4967   4984   5036   5053   5079   5097   5123   5149   5201   5228   5254
                         5280   5306   5333   5360   5387   5414   5440   5466   5492   5518   5544   5570   5597
                         5625   5653   5681   5709   5737   5767   5793   5828   5854   5885   5911   5941   5966
                         5992   6019   6045   6070   6096   6122   6148   6174   6200   6226   6251   6277   6303
                         6329   6355   6380   6406   6432   6458   6484   6510   6536   6562   6588   6616   6645
                         6677   6709   6736   6763   6790   6817   6844   6871   6897   6924   6951   6978   7005
                         7032   7059   7086   7113   7140   7167   7194   7221   7248   7274   7300   7326   7352
                         7378   7405   7427   7453   7474   7501   7527   7553   7574   7601   7628   7652   7676
                         7701   7725   7750   7769   7789   7810   7831   7860   7889   7918   7947   7976   8005
                         8034   8063   8085   8107   8129   8151   8177   8204   8231   8258   8287   8316   8349
                         8378   8405   8432   8459   8486   8513   8542   8571   8599   8626   8653   8680   8710
                         8741   8767   8793   8819   8845   8866   8887   8919   8951   8983   9015   9052   9089
                         9126   9163   9200   9237   9278   9315   9347   9379   9411   9443   9464   9485   9506
                         9527   9549   9570   9597   9625   9645   9664   9688   9712   9733   9760   9787   9814
                         9841   9863   9885   9907   9929   9951   9973   9995  10022  10053  10080  10107  10135
                        10162  10189  10216  10242  10271  10300  10329  10358  10387  10415  10444  10473  10501
                        10528  10555  10583  10610  10637  10663  10690  10717  10744  10773  10802  10832  10862
                        10892  10926  10956  10986  11015  11034  11055  11077  11099  11121  11143  11165  11187
                        11208  11230  11251  11273  11295  11317  11344  11366  11398  11424  11455  11477  11507
                        11533  11559  11585  11615  11645  11673  11704  11733  11758  11787  11820  11852  11885
                        11913  11947  11962  11990  12009  12028  12046  12066  12107  12144  12181  12241  12301
                        12325  12343  12361  12391  12423  12456  12489  12546  12603  12634  12665  12696  12727
                        12758  12794  12829  12874  12957  12977  13017  13055  13080  13105  13130  13154  13193
                        13231  13283  13343  13403  13433  13465  13497  13520  13543  13565  13598  13633  13668
                        13702  13728  13755  13782  13812  13854  13883  13898  13923  13939  13956  14004  14064
                        14124  14184  14230  14274  14304  14334  14365  14401  14437  14473  14509  14540  14564
                        14593  14622  14651  14678  14720  14744  14769  14792  14820  14849  14898  14941  14992
                        15036  15078  15156  15219  15286  15327  15403  15444  15569  15644  15685  15690  15708
                        15714  15716  15763  15922  15923  15924  15925  15926  15943  15955  15957  15958  15959
```

```
                    15966   15967   15968   15980   15983   15990   16004   16005   16006   16007   16027   16031   16043
                    16046   16057
$SWREG  001142       943#    4628
$SWRMK= 000000      15926
$TESTN  001124       934#    4639*  15930   15978*  15988*  17129
$TIMES  001110       921#    4638*  12423*  12456*  15690*  15966*  15973   15976*  15990
$TKB    001046       901#
$TKS    001044       900#
$TMP0   001076       916#   14304*  14334*  14365*  14401*  14437*  14473*  14509*  14540*  14564*  14593*  14622*  14651*
                    14792*  14820*  14853*  14951*  14960   15095*  15125*  17096   17112
$TMP1   001100       917#   14952*  14965*  14971   15012*  15052*  15096*  15126*  17096   17098   17100
$TMP2   001102       918#   14943*  14953   14963   14970*  14972   15013*  15053*  17098   17100
$TMP3   001104       919#   14959*  14963   15014*  17100
$TMP4   001106       920#   14948*  14966   14969*
$TN   = 000770       695#     705   1936    1940    1941#   1944    1949    1953    1954#   1958    1963    1967    1968#
                     1971    1975    1979    1980#   1984    1989    1993    1994#   1999    2004    2008    2009#   2013
                     2018    2022    2023#   2031    2035    2039    2040#   2047    2051    2055    2056#   2063    2068
                     2072    2073#   2085    2090    2094    2095#   2107    2112    2116    2117#   2125    2130    2134
                     2135#   2143    2148    2152    2153#   2162    2167    2171    2172#   2181    2186    2190    2191#
                     2199    2203    2207    2208#   2221    2226    2230    2231#   2244    2249    2253    2254#   2261
                     2266    2270    2271#   2283    2288    2292    2293#   2309    2314    2318    2319#   2338    2344
                     2348    2349#   2358    2363    2367    2368#   2376    2381    2385    2386#   2392    2397    2401
                     2402#   2410    2415    2419    2420#   2429    2434    2438    2439#   2450    2455    2459    2460#
                     2471    2475    2479    2480#   2491    2496    2500    2501#   2512    2517    2521    2522#   2534
                     2539    2543    2544#   2557    2563    2567    2568#   2580    2586    2590    2591#   2603    2609
                     2613    2614#   2630    2634    2635#   2641    2646    2650    2651#   2660    2665    2669    2670#
                     2678    2683    2687    2688#   2701    2706    2710    2711#   2718    2723    2727    2728#   2738
                     2742    2743#   2751    2756    2760    2761#   2768    2773    2777    2778#   2793    2798    2802
                     2803#   2818    2823    2827    2828#   2837    2842    2846    2847#   2862    2866    2870    2871#
                     2884    2889    2893    2894#   2901    2905    2909    2910#   2918    2923    2927    2928#   2944
                     2949    2953    2954#   2972    2977    2981    2982#   2990    2995    2999    3000#   3008    3013
                     3017    3018#   3026    3031    3035    3036#   3043    3048    3052    3053#   3072    3076    3077#
                     3083    3088    3092    3093#   3101    3106    3110    3111#   3120    3125    3129    3130#   3152
                     3157    3161    3162#   3188    3194    3198    3199#   3225    3231    3235    3236#   3262    3268
                     3272    3273#   3299    3305    3309    3310#   3326    3330    3334    3335#   3351    3355    3359
                     3360#   3376    3380    3384    3385#   3401    3406    3410    3411#   3427    3432    3436    3437#
                     3453    3458    3462    3463#   3479    3484    3488    3489#   3505    3510    3514    3515#   3528
                     3532    3536    3537#   3550    3555    3559    3560#   3591    3595    3596#   3627    3631    3632#
                     3641    3645    3649    3650#   3659    3663    3667    3668#   3677    3681    3685    3686#   3699
                     3704    3708    3709#   3715    3720    3724    3725#   3743    3748    3752    3753#   3757    3762
                     3766    3767#   3771    3776    3780    3781#   3785    3790    3794    3795#   3800    3805    3809
                     3810#   3815    3820    3824    3825#   3830    3835    3839    3840#   3844    3849    3853    3854#
                     3859    3864    3868    3869#   3874    3879    3883    3884#   3889    3894    3898    3899#   3904
                     3909    3913    3914#   3919    3924    3928    3929#   3934    3939    3943    3944#   3948    3953
                     3957    3958#   3963    3968    3972    3973#   3978    3983    3987    3988#   3993    3998    4002
                     4003#   4008    4013    4017    4018#   4022    4027    4031    4032#   4037    4042    4046    4047#
                     4052    4057    4061    4062#   4067    4072    4076    4077#   4093    4098    4102    4103#   4111
                     4116    4120    4121#   4128    4133    4137    4138#   4164    4169    4173    4174#   4185    4190
                     4194    4195#   4209    4214    4218    4219#   4240    4245    4249    4250#   4261    4267    4271
                     4272#   4301    4307    4311    4312#   4328    4332    4333#   4362    4368    4372    4373#   4391
                     4395    4396#   4408    4413    4417    4418#   4428    4432    4436    4437#   4456    4460    4461#
                     4472    4477    4481    4482#   4505    4509    4510#   4518    4526    4530    4531#   4538    4544
                     4548    4549#   4556    4562    4574    4575#   4642    4647    4649#   4652    4656    4661    4663#
                     4666    4670    4675    4677#   4686    4690    4695    4697#   4708    4712    4717    4719#   4728
                     4732    4737    4739#   4750    4754    4759    4761#   4764    4768    4773    4775#   4779    4783
                     4788    4790#   4794    4798    4803    4805#   4809    4813    4818    4820#   4824    4828    4833
```

4835#	4838	4842	4847	4849#	4853	4857	4862	4864#	4868	4872	4877	4879#
4883	4887	4892	4894#	4909	4913	4918	4920#	4930	4934	4939	4941#	4956
4960	4965	4967#	4973	4977	4982	4984#	5025	5029	5034	5036#	5042	5046
5051	5053#	5068	5072	5077	5079#	5086	5090	5095	5097#	5112	5116	5121
5123#	5138	5142	5147	5149#	5190	5194	5199	5201#	5217	5221	5226	5228#
5243	5247	5252	5254#	5269	5273	5278	5280#	5295	5299	5304	5306#	5322
5326	5331	5333#	5349	5353	5358	5360#	5376	5380	5385	5387#	5403	5407
5412	5414#	5429	5433	5438	5440#	5455	5459	5464	5466#	5481	5485	5490
5492#	5507	5511	5516	5518#	5533	5537	5542	5544#	5559	5563	5568	5570#
5586	5590	5595	5597#	5613	5618	5623	5625#	5641	5646	5651	5653#	5669
5674	5679	5681#	5697	5702	5707	5709#	5725	5730	5735	5737#	5755	5760
5765	5767#	5781	5786	5791	5793#	5816	5821	5826	5828#	5842	5847	5852
5854#	5873	5878	5883	5885#	5899	5904	5909	5911#	5929	5934	5939	5941#
5954	5959	5964	5966#	5981	5985	5990	5992#	6008	6012	6017	6019#	6034
6038	6043	6045#	6059	6063	6068	6070#	6085	6089	6094	6096#	6111	6115
6120	6122#	6137	6141	6146	6148#	6163	6167	6172	6174#	6189	6193	6198
6200#	6215	6219	6224	6226#	6240	6244	6249	6251#	6266	6270	6275	6277#
6292	6296	6301	6303#	6318	6322	6327	6329#	6344	6348	6353	6355#	6369
6373	6378	6380#	6395	6399	6404	6406#	6421	6425	6430	6432#	6447	6451
6456	6458#	6473	6477	6482	6484#	6499	6503	6508	6510#	6525	6529	6534
6536#	6551	6555	6560	6562#	6577	6581	6586	6588#	6604	6609	6614	6616#
6633	6638	6643	6645#	6665	6670	6675	6677#	6697	6702	6707	6709#	6724
6729	6734	6736#	6751	6756	6761	6763#	6778	6783	6788	6790#	6805	6810
6815	6817#	6832	6837	6842	6844#	6859	6864	6869	6871#	6885	6890	6895
6897#	6912	6917	6922	6924#	6939	6944	6949	6951#	6966	6971	6976	6978#
6993	6998	7003	7005#	7020	7025	7030	7032#	7047	7052	7057	7059#	7074
7079	7084	7086#	7101	7106	7111	7113#	7128	7133	7138	7140#	7155	7160
7165	7167#	7182	7187	7192	7194#	7209	7214	7219	7221#	7236	7241	7246
7248#	7263	7267	7272	7274#	7289	7293	7298	7300#	7315	7319	7324	7326#
7341	7345	7350	7352#	7367	7371	7376	7378#	7393	7398	7403	7405#	7415
7420	7425	7427#	7441	7446	7451	7453#	7462	7467	7472	7474#	7489	7494
7499	7501#	7515	7520	7525	7527#	7541	7546	7551	7553#	7562	7567	7572
7574#	7590	7594	7599	7601#	7617	7621	7626	7628#	7641	7645	7650	7652#
7665	7669	7674	7676#	7690	7694	7699	7701#	7714	7718	7723	7725#	7739
7743	7748	7750#	7758	7762	7767	7769#	7778	7782	7787	7789#	7798	7803
7808	7810#	7819	7824	7829	7831#	7849	7853	7858	7860#	7878	7882	7887
7889#	7907	7911	7916	7918#	7936	7940	7945	7947#	7965	7969	7974	7976#
7994	7998	8003	8005#	8023	8027	8032	8034#	8052	8056	8061	8063#	8073
8078	8083	8085#	8095	8100	8105	8107#	8117	8122	8127	8129#	8139	8144
8149	8151#	8166	8170	8175	8177#	8193	8197	8202	8204#	8220	8224	8229
8231#	8247	8251	8256	8258#	8275	8280	8285	8287#	8304	8309	8314	8316#
8337	8342	8347	8349#	8366	8371	8376	8378#	8394	8398	8403	8405#	8421
8425	8430	8432#	8448	8452	8457	8459#	8475	8479	8484	8486#	8502	8506
8511	8513#	8530	8535	8540	8542#	8559	8564	8569	8571#	8587	8592	8597
8599#	8615	8619	8624	8626#	8642	8646	8651	8653#	8669	8673	8678	8680#
8698	8703	8708	8710#	8729	8734	8739	8741#	8756	8760	8765	8767#	8782
8786	8791	8793#	8808	8812	8817	8819#	8834	8838	8843	8845#	8854	8859
8864	8866#	8875	8880	8885	8887#	8906	8912	8917	8919#	8938	8944	8949
8951#	8970	8976	8981	8983#	9002	9008	9013	9015#	9039	9045	9050	9052#
9076	9082	9087	9089#	9113	9119	9124	9126#	9150	9156	9161	9163#	9187
9193	9198	9200#	9224	9230	9235	9237#	9265	9271	9276	9278#	9302	9308
9313	9315#	9334	9340	9345	9347#	9366	9372	9377	9379#	9398	9404	9409
9411#	9430	9436	9441	9443#	9452	9457	9462	9464#	9473	9478	9483	9485#
9494	9499	9504	9506#	9515	9520	9525	9527#	9536	9542	9547	9549#	9558
9563	9568	9570#	9586	9590	9595	9597#	9614	9618	9623	9625#	9634	9638
9643	9645#	9653	9657	9662	9664#	9677	9681	9686	9688#	9701	9705	9710

9712#	9721	9726	9731	9733#	9749	9753	9758	9760#	9775	9780	9785	9787#
9802	9807	9812	9814#	9829	9834	9839	9841#	9851	9856	9861	9863#	9873
9878	9883	9885#	9895	9900	9905	9907#	9917	9922	9927	9929#	9939	9944
9949	9951#	9961	9966	9971	9973#	9983	9988	9993	9995#	10011	10015	10020
10022#	10042	10046	10051	10053#	10069	10073	10078	10080#	10096	10100	10105	10107#
10123	10128	10133	10135#	10151	10155	10160	10162#	10178	10182	10187	10189#	10205
10209	10214	10216#	10231	10235	10240	10242#	10259	10264	10269	10271#	10288	10293
10298	10300#	10317	10322	10327	10329#	10346	10351	10356	10358#	10375	10380	10385
10387#	10404	10408	10413	10415#	10432	10437	10442	10444#	10461	10466	10471	10473#
10489	10494	10499	10501#	10517	10521	10526	10528#	10544	10548	10553	10555#	10572
10576	10581	10583#	10599	10603	10608	10610#	10626	10630	10635	10637#	10653	10656
10661	10663#	10679	10683	10688	10690#	10706	10710	10715	10717#	10733	10737	10742
10744#	10761	10766	10771	10773#	10790	10795	10800	10802#	10820	10825	10830	10832#
10850	10855	10860	10862#	10880	10885	10890	10892#	10914	10919	10924	10926#	10944
10949	10954	10956#	10974	10979	10984	10986#	11003	11008	11013	11015#	11023	11027
11032	11034#	11043	11048	11053	11055#	11065	11070	11075	11077#	11087	11092	11097
11099#	11109	11114	11119	11121#	11131	11136	11141	11143#	11153	11158	11163	11165#
11175	11180	11185	11187#	11196	11201	11206	11208#	11218	11223	11228	11230#	11239
11244	11249	11251#	11261	11266	11271	11273#	11279	11284	11288	11293	11295#	11301
11306	11310	11315	11317#	11323	11333	11337	11342	11344#	11350	11355	11359	11364
11366#	11372	11382	11385	11391	11396	11398#	11404	11409	11412	11417	11422	11424#
11430	11434	11444	11448	11453	11455#	11461	11466	11470	11475	11477#	11483	11493
11496	11500	11505	11507#	11513	11518	11521	11526	11531	11533#	11539	11544	11547
11552	11557	11559#	11565	11570	11573	11578	11583	11585#	11591	11594	11599	11602
11608	11613	11615#	11621	11624	11629	11632	11638	11643	11645#	11657	11666	11671
11673#	11697	11702	11704#	11726	11731	11733#	11751	11756	11758#	11770	11780	11785
11787#	11799	11813	11818	11820#	11835	11845	11850	11852#	11864	11878	11883	11885#
11897	11906	11911	11913#	11926	11940	11945	11947#	11951	11956	11961	11962#	11977
11980	11983	11988	11990#	11998	12002	12007	12009#	12017	12021	12026	12028#	12036
12040	12045	12046#	12055	12059	12064	12066#	12092	12100	12105	12107#	12129	12137
12142	12144#	12166	12174	12179	12181#	12193	12201	12213	12222	12234	12239	12241#
12253	12261	12273	12282	12294	12299	12301#	12318	12323	12325#	12331	12336	12341
12343#	12349	12354	12359	12361#	12384	12389	12391#	12416	12421	12423#	12449	12454
12456#	12482	12487	12489#	12539	12544	12546#	12596	12601	12603#	12627	12632	12634#
12658	12663	12665#	12689	12694	12696#	12720	12725	12727#	12751	12756	12758#	12787
12792	12794#	12819	12827	12829#	12863	12872	12874#	12950	12955	12957#	12970	12975
12977#	13010	13015	13017#	13048	13053	13055#	13073	13078	13080#	13098	13103	13105#
13123	13128	13130#	13147	13152	13154#	13181	13186	13191	13193#	13219	13224	13229
13231#	13258	13276	13281	13283#	13327	13336	13341	13343#	13387	13396	13401	13403#
13426	13431	13433#	13458	13463	13465#	13490	13495	13497#	13513	13518	13520#	13536
13541	13543#	13558	13563	13565#	13589	13596	13598#	13612	13626	13631	13633#	13647
13661	13666	13668#	13682	13696	13701	13702#	13722	13727	13728#	13749	13754	13755#
13776	13781	13782#	13806	13811	13812#	13846	13853	13854#	13876	13881	13883#	13891
13896	13898#	13916	13921	13923#	13932	13937	13939#	13949	13954	13956#	13969	14003
14004#	14029	14063	14064#	14089	14123	14124#	14149	14183	14184#	14209	14229	14230#
14245	14254	14273	14274#	14289	14298	14303	14304#	14324	14328	14333	14334#	14355
14359	14364	14365#	14391	14395	14400	14401#	14427	14431	14436	14437#	14463	14467
14472	14473#	14499	14503	14508	14509#	14530	14534	14539	14540#	14554	14558	14563
14564#	14583	14587	14592	14593#	14612	14616	14621	14622#	14641	14645	14650	14651#
14663	14667	14676	14678#	14706	14713	14718	14720#	14734	14737	14742	14744#	14759
14762	14767	14769#	14783	14786	14791	14792#	14811	14814	14819	14820#	14840	14843
14848	14849#	14872	14888	14896	14898#	14926	14939	14941#	14977	14990	14992#	15023
15034	15036#	15058	15076	15078#	15145	15154	15156#	15206	15217	15219#	15276	15284
15286#	15313	15325	15327#	15390	15401	15403#	15429	15442	15444#	15551	15567	15569#
15638	15642	15644#	15675									

$1PB 001052 903# 16185* 16196

```
$TPFLG  001057            907#  16134   16196
$TPS    001050            902#  16183   16196
$TRAP   063166           4405    4622   16341#
$TRAP2  063210          16352#  16363
$TRP  = 000005          16356#  16365#  16366#  16367#  16368#
$TRPAD  063222          16346   16363#
$TSTM   000704            857#
$TSTNM  001002            880#  15921   15935*  15977*  15978   15983   15985   15991   16026   16057
$TYPBN= ****** U        16368
$TYPDS= ****** U        16368
$TYPE   062210          16134#  16306   16356   16364
$TYPEC  062422          16164   16171   16178   16183#  16184
$TYPEX  062470          16189   16191   16194#
$TYPOC  062516          16227#  16365
$TYPON  062532          16226   16229#  16367
$TYPOS  062472          16222#  16366
$UNIT   001132            937#
$UNITM  000710            859#
$USWR   001144            944#
$XTSTR  061350          15946#
$$GET4= 000000          15708#
$OFILL  062715          16223*  16227*  16237   16272#
$40CAT= ****** U        15943   16033
.     = 070034            829#    833#    838#    844     845#    847#    849#    865     866#    868#    870#    877#    926
                         2295    2736    2956    3688    4197    4922    5795    6647    7503    8318    9239   10024   10894
                        11706   12405   12616   12647   12678   12709   12740   12760   12775   12807   13900   14851   14923
                        15716   15717#  15738   15765   15990   15991   16057   16107#  16196   16327#  16391#  16575   16576
                        16598#  17117#
.$ASTA= ****** U        16279   16282
.$X   = 000700            844#    849
```

COMMEN 1# 827#
ENDCOM 1# 827#
ENDERR 15997# 16053
ENDPAS 15680# 15699
ENDSCO 15915# 15985
ERROR 721# 4425 4654 4668 4688 4704 4710 4730 4746 4752 4766 4781 4796 4811 4826
4840 4855 4870 4885 4906 4911 4932 4952 4958 4975 4997 5002 5017 5022 5027
5044 5065 5070 5088 5109 5114 5135 5140 5162 5167 5182 5187 5192 5214 5219
5240 5245 5266 5271 5292 5297 5319 5324 5346 5351 5373 5378 5400 5405 5426
5431 5452 5457 5478 5483 5504 5509 5530 5535 5556 5561 5583 5588 5610 5616
5638 5644 5666 5672 5694 5700 5722 5728 5747 5752 5758 5778 5784 5808 5813
5819 5839 5845 5865 5870 5876 5896 5902 5921 5926 5932 5951 5957 5978 5983
6005 6010 6031 6036 6056 6061 6082 6087 6108 6113 6134 6139 6160 6165 6186
6191 6212 6217 6237 6242 6263 6268 6289 6294 6315 6320 6341 6346 6367 6371
6392 6397 6418 6423 6444 6449 6470 6475 6496 6501 6522 6527 6548 6553 6574
6579 6601 6607 6630 6636 6662 6668 6689 6694 6700 6722 6727 6749 6754 6776
6781 6803 6808 6830 6835 6857 6862 6883 6888 6910 6915 6937 6942 6964 6969
6991 6996 7018 7023 7045 7050 7072 7077 7099 7104 7126 7131 7153 7158 7180
7185 7207 7212 7234 7239 7260 7265 7286 7291 7312 7317 7338 7343 7364 7369
7390 7396 7418 7438 7444 7465 7486 7492 7518 7538 7544 7565 7587 7592 7614
7619 7638 7643 7662 7667 7687 7692 7711 7716 7736 7741 7760 7780 7801 7822
7845 7851 7874 7880 7903 7909 7932 7938 7961 7967 7990 7996 8019 8025 8048
8054 8076 8098 8120 8142 8163 8168 8190 8195 8217 8222 8244 8249 8272 8278
8301 8307 8334 8340 8363 8369 8391 8396 8418 8423 8445 8450 8472 8477 8499
8504 8527 8533 8556 8562 8584 8590 8612 8617 8639 8644 8666 8671 8695 8701
8726 8732 8753 8758 8779 8784 8805 8810 8831 8836 8857 8878 8903 8910 8935
8942 8967 8974 8999 9006 9031 9036 9043 9068 9073 9080 9105 9110 9117 9142
9147 9154 9179 9184 9191 9216 9221 9228 9257 9262 9269 9294 9299 9306 9331
9338 9363 9370 9395 9402 9427 9434 9455 9476 9497 9518 9539 9561 9583 9588
9611 9616 9636 9655 9674 9679 9698 9703 9724 9746 9751 9772 9778 9799 9805
9826 9832 9854 9876 9898 9920 9942 9964 9986 10008 10013 10039 10044 10066 10071
10093 10098 10120 10126 10148 10153 10175 10180 10202 10207 10228 10233 10256 10262 10285
10291 10314 10320 10343 10349 10372 10378 10401 10407 10429 10435 10458 10464 10486 10492
10514 10519 10541 10546 10569 10574 10596 10601 10623 10628 10650 10655 10676 10681 10703
10708 10730 10735 10758 10764 10787 10793 10817 10823 10847 10853 10877 10883 10911 10917
10941 10947 10971 10977 11000 11006 11025 11046 11068 11090 11112 11134 11156 11178 11199
11221 11242 11264 11278 11286 11300 11308 11322 11330 11335 11349 11357 11371 11379 11384
11387 11403 11411 11415 11429 11433 11441 11446 11460 11468 11482 11490 11495 11498 11512
11520 11524 11538 11546 11549 11564 11572 11575 11590 11593 11601 11606 11620 11623 11631
11636 11653 11663 11683 11688 11693 11716 11723 11741 11747 11766 11776 11795 11805 11809
11828 11831 11841 11860 11870 11874 11893 11903 11921 11932 11936 11950 11979 11981 12000
12019 12038 12057 12076 12084 12089 12096 12119 12126 12133 12156 12163 12170 12191 12199
12212 12220 12230 12251 12259 12272 12280 12290 12313 12334 12352 12377 12408 12436 12444
12469 12477 12512 12519 12528 12531 12569 12576 12585 12588 12619 12650 12681 12712 12743
12779 12811 12853 12857 12912 12919 12926 12932 12937 12965 12990 12998 13005 13029 13037
13044 13068 13093 13118 13141 13162 13170 13178 13201 13209 13216 13239 13247 13255 13272
13294 13304 13314 13324 13354 13364 13374 13384 13412 13422 13444 13453 13476 13485 13508
13531 13554 13576 13583 13611 13621 13646 13656 13681 13691 13714 13740 13767 13795 13824
13871 13889 13913 13930 13947 13967 14022 14082 14142 14202 14243 14287 14319 14326 14350
14357 14381 14388 14393 14417 14424 14429 14453 14460 14465 14489 14496 14501 14525 14532
14556 14578 14585 14607 14614 14636 14643 14665 14711 14731 14735 14756 14760 14780 14784
14806 14812 14835 14841 14867 14873 14907 14911 14924 14968 15015 15054 15097 15106 15127
15164 15174 15179 15198 15241 15270 15297 15308 15340 15347 15355 15369 15377 15385 15407
15418 15424 15461 15477 15486 15497 15508 15523 15531 15545 15593 15604 15627 15657 15672
15933
ESCAPE 1# 827#

```
GETPR1      1#    827#
GETSWR      1#    827#
MSG      4562#   4564   12819#  12821   12863#  12865   13589#  13591   13846#  13848   13969#  13971   14029#  14031   14089#
        14091   14149#  14151   14209#  14211   14254#  14256   14667#  14669   15144#  15147   15205#  15208   15275#  15278
        15312#  15315   15389#  15392   15428#  15431   15550#  15553
MSGJ    14888#  14890
MSGM1   14925#  14928
MSGM10  15022#  15025
MSGM11  15057#  15060
MSGM3   14977#  14979
MULT        1#    827#
NEWTST      1#    827#  1936    1949    1963    1975    1989    2004    2018    2035    2051    2068    2090    2112    2130
         2148    2167   2186    2203    2226    2249    2266    2288    2314    2344    2363    2381    2397    2415    2434
         2455    2475   2496    2517    2539    2563    2586    2609    2630    2646    2665    2683    2706    2723    2738
         2756    2773   2798    2823    2842    2866    2889    2905    2923    2949    2977    2995    3013    3031    3048
         3072    3088   3106    3125    3157    3194    3231    3268    3305    3330    3355    3380    3406    3432    3458
         3484    3510   3532    3555    3591    3627    3645    3663    3681    3704    3720    3748    3762    3776    3790
         3805    3820   3835    3849    3864    3879    3894    3909    3924    3939    3953    3968    3983    3998    4013
         4027    4042   4057    4072    4098    4116    4133    4169    4190    4214    4245    4267    4307    4328    4368
         4391    4413   4432    4456    4477    4505    4526    4544    4562    4642    4656    4670    4690    4712    4732
         4754    4768   4783    4798    4813    4828    4842    4857    4872    4887    4913    4934    4960    4977    5029
         5046    5072   5090    5116    5142    5194    5221    5247    5273    5299    5326    5353    5380    5407    5433
         5459    5485   5511    5537    5563    5590    5618    5646    5674    5702    5730    5760    5786    5821    5847
         5878    5904   5934    5959    5985    6012    6038    6063    6089    6115    6141    6167    6193    6219    6244
         6270    6296   6322    6348    6373    6399    6425    6451    6477    6503    6529    6555    6581    6609    6638
         6670    6702   6729    6756    6783    6810    6837    6864    6890    6917    6944    6971    6998    7025    7052
         7079    7106   7133    7160    7187    7214    7241    7267    7293    7319    7345    7371    7398    7420    7446
         7467    7494   7520    7546    7567    7594    7621    7645    7669    7694    7718    7743    7762    7782    7803
         7824    7853   7882    7911    7940    7969    7998    8027    8056    8078    8100    8122    8144    8170    8197
         8224    8251   8280    8309    8342    8371    8398    8425    8452    8479    8506    8535    8564    8592    8619
         8646    8673   8703    8734    8760    8786    8812    8838    8859    8880    8912    8944    8976    9008    9045
         9082    9119   9156    9193    9230    9271    9308    9340    9372    9404    9436    9457    9478    9499    9520
         9542    9563   9590    9618    9638    9657    9681    9705    9726    9753    9780    9807    9834    9856    9878
         9900    9922   9944    9966    9988   10015   10046   10073   10100   10128   10155   10182   10209   10235   10264
        10293   10322  10351   10380   10408   10437   10466   10494   10521   10548   10576   10603   10630   10656   10683
        10710   10737  10766   10795   10825   10855   10885   10919   10949   10979   11008   11027   11048   11070   11092
        11114   11136  11158   11180   11201   11223   11244   11266   11288   11310   11337   11359   11391   11417   11448
        11470   11500  11526   11552   11578   11608   11638   11666   11697   11726   11751   11780   11813   11845   11878
        11906   11940  11956   11983   12002   12021   12040   12059   12100   12137   12174   12234   12294   12318   12336
        12354   12384  12416   12449   12482   12539   12596   12627   12658   12689   12720   12751   12787   12819   12863
        12950   12970  13010   13048   13073   13098   13123   13147   13186   13224   13276   13336   13396   13426   13458
        13490   13513  13536   13558   13589   13626   13661   13696   13722   13749   13776   13806   13846   13876   13891
        13916   13932  13949   13969   14029   14089   14149   14209   14254   14298   14328   14359   14395   14431   14467
        14503   14534  14558   14587   14616   14645   14667   14713   14737   14762   14786   14814   14843   14888   14926
        14977   15023  15058   15145   15206   15276   15313   15390   15429   15551   15638
NOINST   1263#  11959  12043   13699   13725   13752   13779   13809   14001   14061   14121   14181   14227   14271   14301
        14331   14362  14398   14434   14470   14506   14527   14561   14590   14619   14648   14789   14817   14846
NOSCOP   1263#   1939   1952    1966    1978    1992    2007    2021    2038    2054    2071    2093    2115    2133    2151
         2170    2189   2206    2229    2252    2269    2291    2317    2347    2366    2384    2400    2418    2437    2458
         2478    2499   2520    2542    2566    2589    2612    2633    2649    2668    2686    2709    2726    2741    2759
         2776    2801   2825    2845    2869    2892    2908    2926    2952    2980    2998    3016    3034    3051    3075
         3091    3109   3128    3160    3197    3234    3271    3308    3333    3358    3383    3409    3435    3461    3487
         3513    3535   3558    3594    3630    3648    3666    3684    3707    3723    3751    3765    3779    3793    3808
         3823    3838   3852    3867    3882    3897    3912    3927    3942    3956    3971    3986    4001    4016    4030
         4045    4060   4075    4101    4119    4136    4172    4193    4217    4248    4270    4310    4331    4371    4394
         4416    4435   4459    4480    4508    4529    4547    4573   13852
```

```
POP          1#    827#  15750  15751  16321  16322
PREERR   15996#  16012
PRENEW    1263#  14895  14938  14989  15033  15075  15153  15216  15283  15324  15400  15441  15566  15641
PRESCO   15914#  15930
PUSH         1#    827#  15728  15734  16282  16284  16305
REPORT       1#    827#
SCOPE      722#   4319   4646   4660   4674   4694   4716   4736   4758   4772   4787   4802   4817   4832   4846
          4861   4876   4891   4917   4938   4964   4981   5033   5050   5076   5094   5120   5146   5198   5225
          5251   5277   5303   5330   5357   5384   5411   5437   5463   5489   5515   5541   5567   5594   5622
          5650   5678   5706   5734   5764   5790   5825   5851   5882   5908   5938   5963   5989   6016   6042
          6067   6093   6119   6145   6171   6197   6223   6248   6274   6300   6326   6352   6377   6403   6429
          6455   6481   6507   6533   6559   6585   6613   6642   6674   6706   6733   6760   6787   6814   6841
          6868   6894   6921   6948   6975   7002   7029   7056   7083   7110   7137   7164   7191   7218   7245
          7271   7297   7323   7349   7375   7402   7424   7450   7471   7498   7524   7550   7571   7598   7625
          7649   7673   7698   7722   7747   7766   7786   7807   7828   7857   7886   7915   7944   7973   8002
          8031   8060   8082   8104   8126   8148   8174   8201   8228   8255   8284   8313   8346   8375   8402
          8429   8456   8483   8510   8539   8568   8596   8623   8650   8677   8707   8738   8764   8790   8816
          8842   8863   8884   8916   8948   8980   9012   9049   9086   9123   9160   9197   9234   9275   9312
          9344   9376   9408   9440   9461   9482   9503   9524   9546   9567   9594   9622   9642   9661   9685
          9709   9730   9757   9784   9811   9838   9860   9882   9904   9926   9948   9970   9992  10019  10050
         10077  10104  10132  10159  10186  10213  10239  10268  10297  10326  10355  10384  10412  10441  10470
         10498  10525  10552  10580  10607  10634  10660  10687  10714  10741  10770  10799  10829  10859  10889
         10923  10953  10983  11012  11031  11052  11074  11096  11118  11140  11162  11184  11205  11227  11248
         11270  11292  11314  11341  11363  11395  11421  11452  11474  11504  11530  11556  11582  11612  11642
         11670  11701  11730  11755  11784  11817  11849  11882  11910  11944  11960  11987  12006  12025  12044
         12063  12104  12141  12178  12238  12298  12322  12340  12358  12388  12420  12453  12486  12543  12600
         12631  12662  12693  12724  12755  12791  12826  12871  12954  12974  13014  13052  13077  13102  13127
         13151  13190  13228  13280  13340  13400  13430  13462  13494  13517  13540  13562  13595  13630  13665
         13700  13726  13753  13780  13810  13837  13880  13895  13920  13936  13953  14002  14062  14122  14182
         14228  14272  14302  14332  14363  14399  14435  14471  14507  14538  14562  14591  14620  14649  14675
         14717  14741  14766  14790  14818  14847  14897  14940  14991  15035  15077  15155  15218  15285  15326
         15402  15443  15568  15643  15689
SETPRI       1#    827#
SETTRA   16356#  16365  16366  16367
SETUP        1#    827#
SKIP         1#    827#  1943   1958   1970   1984   1999   2012   2031   2047   2063   2085   2107   2125   2143
          2162   2181   2199   2221   2244   2261   2283   2309   2338   2358   2376   2392   2410   2429   2450
          2471   2491   2512   2534   2557   2580   2603   2641   2660   2678   2701   2718   2751   2768   2793
          2818   2837   2862   2884   2901   2918   2944   2972   2990   3008   3026   3043   3083   3101   3120
          3152   3188   3225   3262   3299   3326   3351   3376   3401   3427   3453   3479   3505   3528   3550
          3641   3659   3677   3699   3715   3742   3757   3770   3784   3800   3815   3829   3844   3858   3873
          3889   3904   3919   3934   3947   3963   3978   3993   4007   4021   4037   4052   4067   4093   4111
          4128   4164   4185   4209   4240   4261   4301   4362   4408   4428   4472   4518   4538   4556   4652
          4665   4686   4708   4728   4750   4764   4778   4793   4808   4824   4838   4852   4867   4882   4909
          4930   4956   4973   5025   5042   5068   5086   5112   5138   5190   5217   5243   5269   5295   5322
          5349   5376   5403   5429   5455   5481   5507   5533   5559   5586   5613   5641   5669   5697   5725
          5755   5781   5816   5842   5873   5899   5929   5954   5981   6008   6034   6059   6085   6111   6137
          6163   6189   6215   6240   6266   6292   6318   6344   6369   6395   6421   6447   6473   6499   6525
          6551   6577   6604   6633   6665   6697   6724   6751   6778   6805   6832   6859   6885   6912   6939
          6966   6993   7020   7047   7074   7101   7128   7155   7182   7209   7236   7263   7289   7315   7341
          7367   7393   7415   7441   7462   7489   7515   7541   7562   7590   7617   7641   7665   7690   7714
          7739   7758   7778   7798   7819   7849   7878   7907   7936   7965   7994   8023   8052   8073   8095
          8117   8139   8166   8193   8220   8247   8275   8304   8337   8366   8394   8421   8448   8475   8502
          8530   8559   8587   8615   8642   8669   8698   8729   8756   8782   8808   8834   8854   8875   8906
          8938   8970   9002   9039   9076   9113   9150   9187   9224   9265   9302   9334   9366   9398   9430
          9452   9473   9494   9515   9536   9558   9586   9614   9634   9653   9677   9701   9721   9749   9775
```

```
                9802    9829    9851    9873    9895    9917    9939    9961    9983   10011   10042   10069   10096   10123   10151
               10178   10205   10231   10259   10288   10317   10346   10375   10404   10432   10461   10489   10517   10544   10572
               10599   10626   10653   10679   10706   10733   10761   10790   10820   10850   10880   10914   10944   10974   11003
               11023   11043   11065   11087   11109   11131   11153   11175   11196   11218   11239   11261   11279   11284   11301
               11306   11323   11333   11350   11355   11372   11382   11385   11404   11409   11412   11430   11434   11444   11461
               11466   11483   11493   11496   11513   11518   11521   11539   11544   11547   11565   11570   11573   11591   11594
               11599   11602   11621   11624   11629   11632   11657   11770   11799   11835   11864   11897   11926   11951   11977
               11980   11998   12017   12036   12055   12092   12129   12166   12193   12201   12213   12222   12253   12261   12273
               12282   12331   12349   13181   13219   13258   13327   13387   13612   13647   13682   14245   14289   14324   14355
               14391   14427   14463   14499   14530   14554   14583   14612   14641   14663   14706   14734   14759   14783   14811
               14840   14872
SLASH            1#    827#
SPACE          827#
STARS            1#    827#    841     843     850     863     873     926     929    1936    1938    1949    1951    1963    1965
                1975    1977    1989    1991    2004    2006    2018    2020    2035    2037    2051    2053    2068    2070    2090
                2092    2112    2114    2130    2132    2148    2150    2167    2169    2186    2188    2203    2205    2226    2228
                2249    2251    2266    2268    2288    2290    2314    2316    2344    2346    2363    2365    2381    2383    2397
                2399    2415    2417    2434    2436    2455    2457    2475    2477    2496    2498    2517    2519    2539    2541
                2563    2565    2586    2588    2609    2611    2630    2632    2646    2648    2665    2667    2683    2685    2706
                2708    2723    2725    2738    2740    2756    2758    2773    2775    2798    2800    2823    2825    2842    2844
                2866    2868    2889    2891    2905    2907    2923    2925    2949    2951    2977    2979    2995    2997    3013
                3015    3031    3033    3048    3050    3072    3074    3088    3090    3106    3108    3125    3127    3157    3159
                3194    3196    3231    3233    3268    3270    3305    3307    3330    3332    3355    3357    3380    3382    3406
                3408    3432    3434    3458    3460    3484    3486    3510    3512    3532    3534    3555    3557    3591    3593
                3627    3629    3645    3647    3663    3665    3681    3683    3704    3706    3720    3722    3748    3750    3762
                3764    3776    3778    3790    3792    3805    3807    3820    3822    3835    3837    3849    3851    3864    3866
                3879    3881    3894    3896    3909    3911    3924    3926    3939    3941    3953    3955    3968    3970    3983
                3985    3998    4000    4013    4015    4027    4029    4042    4044    4057    4059    4072    4074    4098    4100
                4116    4118    4133    4135    4169    4171    4190    4192    4214    4216    4245    4247    4267    4269    4307
                4309    4328    4330    4368    4370    4391    4393    4413    4415    4432    4434    4456    4458    4477    4479
                4505    4507    4526    4528    4544    4546    4562    4572    4642    4644    4656    4658    4670    4672    4690
                4692    4712    4714    4732    4734    4754    4756    4768    4770    4783    4785    4798    4800    4813    4815
                4828    4830    4842    4844    4857    4859    4872    4874    4887    4889    4913    4915    4934    4936    4960
                4962    4977    4979    5029    5031    5046    5048    5072    5074    5090    5092    5116    5118    5142    5144
                5194    5196    5221    5223    5247    5249    5273    5275    5299    5301    5326    5328    5353    5355    5380
                5382    5407    5409    5433    5435    5459    5461    5485    5487    5511    5513    5537    5539    5563    5565
                5590    5592    5618    5620    5646    5648    5674    5676    5702    5704    5730    5732    5760    5762    5786
                5788    5821    5823    5847    5849    5878    5880    5904    5906    5934    5936    5959    5961    5985    5987
                6012    6014    6038    604C    6063    6065    6089    6091    6115    6117    6141    6143    6167    6169    6193
                6195    6219    6221    6244    6246    6270    6272    6296    6298    6322    6324    6348    6350    6373    6375
                6399    6401    6425    6427    6451    6453    6477    6479    6503    6505    6529    6531    6555    6557    6581
                6583    6609    6611    6638    6640    6670    6672    6702    6704    6729    6731    6756    6758    6783    6785
                6810    6812    6837    6839    6864    6866    6890    6892    6917    6919    6944    6946    6971    6973    6998
                7000    7025    7027    7052    7054    7079    7081    7106    7108    7133    7135    7160    7162    7187    7189
                7214    7216    7241    7243    7267    7269    7293    7295    7319    7321    7345    7347    7371    7373    7398
                7400    7420    7422    7446    7448    7467    7469    7494    7496    7520    7522    7546    7548    7567    7569
                7594    7596    7621    7623    7645    7647    7669    7671    7694    7696    7718    7720    7743    7745    7762
                7764    7782    7784    7803    7805    7824    7826    7853    7855    7882    7884    7911    7913    7940    7942
                7969    7971    7998    8000    8027    8029    8056    8058    8078    8080    8100    8102    8122    8124    8144
                8146    8170    8172    8197    8199    8224    8226    8251    8253    8280    8282    8309    8311    8342    8344
                8371    8373    8398    8400    8425    8427    8452    8454    8479    8481    8506    8508    8535    8537    8564
                8566    8592    8594    8619    8621    8646    8648    8673    8675    8703    8705    8734    8736    8760    8762
                8786    8788    8812    8814    8838    8840    8859    8861    8880    8882    8912    8914    8944    8946    8976
                8978    9008    9010    9045    9047    9082    9084    9119    9121    9156    9158    9193    9195    9230    9232
                9271    9273    9308    9310    9340    9342    9372    9374    9404    9406    9436    9438    9457    9459    9478
                9480    9499    9501    9520    9522    9542    9544    9563    9565    9590    9592    9618    9620    9638    9640
```

```
          9657    9659    9681    9683    9705    9707    9726    9728    9753    9755    9780    9782    9807    9809    9834
          9836    9856    9858    9878    9880    9900    9902    9922    9924    9944    9946    9966    9968    9988    9990
         10015   10017   10046   10048   10073   10075   10100   10102   10128   10130   10155   10157   10182   10184   10209
         10211   10235   10237   10264   10266   10293   10295   10322   10324   10351   10353   10380   10382   10408   10410
         10437   10439   10466   10468   10494   10496   10521   10523   10548   10550   10576   10578   10603   10605   10630
         10632   10656   10658   10683   10685   10710   10712   10737   10739   10766   10768   10795   10797   10825   10827
         10855   10857   10885   10887   10919   10921   10949   10951   10979   10981   11008   11010   11027   11029   11048
         11050   11070   11072   11092   11094   11114   11116   11136   11138   11158   11160   11180   11182   11201   11203
         11223   11225   11244   11246   11266   11268   11288   11290   11310   11312   11337   11339   11359   11361   11391
         11393   11417   11419   11448   11450   11470   11472   11500   11502   11526   11528   11552   11554   11578   11580
         11608   11610   11638   11640   11666   11668   11697   11699   11726   11728   11751   11753   11780   11782   11813
         11815   11845   11847   11878   11880   11906   11908   11940   11942   11956   11958   11983   11985   12002   12004
         12021   12023   12040   12042   12059   12061   12100   12102   12137   12139   12174   12176   12234   12236   12294
         12296   12318   12320   12336   12338   12354   12356   12384   12386   12416   12418   12449   12451   12482   12484
         12539   12541   12596   12598   12627   12629   12658   12660   12689   12691   12720   12722   12751   12753   12787
         12789   12819   12824   12863   12869   12950   12952   12970   12972   13010   13012   13048   13050   13073   13075
         13098   13100   13123   13125   13147   13149   13186   13188   13224   13226   13276   13278   13336   13338   13396
         13398   13426   13428   13458   13460   13490   13492   13513   13515   13536   13538   13558   13560   13589   13593
         13626   13628   13661   13663   13696   13698   13722   13724   13749   13751   13776   13778   13806   13808   13846
         13851   13876   13878   13891   13893   13916   13918   13932   13934   13949   13951   13969   14000   14029   14060
         14089   14120   14149   14180   14209   14226   14254   14270   14298   14300   14328   14330   14359   14361   14395
         14397   14431   14433   14467   14469   14503   14505   14534   14536   14558   14560   14587   14589   14616   14618
         14645   14647   14667   14673   14713   14715   14737   14739   14762   14764   14786   14788   14814   14816   14843
         14845   14888   14894   14926   14937   14977   14988   15023   15032   15058   15074   15145   15152   15206   15215
         15276   15282   15313   15323   15390   15399   15429   15440   15551   15565   15638   15640   15683   15724   15740
         15784   15918   16000   16063   16119   16199   16277   16335
SWRSU       1#    827#
TESTNO   1263#   1940    1953    1967    1979    1993    2008    2022    2039    2055    2072    2094    2116    2134    2152
          2171    2190    2207    2230    2253    2270    2292    2318    2348    2367    2385    2401    2419    2438    2459
          2479    2500    2521    2543    2567    2590    2613    2634    2650    2669    2687    2710    2727    2742    2760
          2777    2802    2827    2846    2870    2893    2909    2927    2953    2981    2999    3017    3035    3052    3076
          3092    3110    3129    3161    3198    3235    3272    3309    3334    3359    3384    3410    3436    3462    3488
          3514    3536    3559    3595    3631    3649    3667    3685    3708    3724    3752    3766    3780    3794    3809
          3824    3839    3853    3868    3883    3898    3913    3928    3943    3957    3972    3987    4002    4017    4031
          4046    4061    4076    4102    4120    4137    4173    4194    4218    4249    4271    4311    4332    4372    4395
          4417    4436    4460    4481    4509    4530    4548    4574    4647    4661    4675    4695    4717    4737    4759
          4773    4788    4803    4818    4833    4847    4862    4877    4892    4918    4939    4965    4982    5034    5051
          5077    5095    5121    5147    5199    5226    5252    5278    5304    5331    5358    5385    5412    5438    5464
          5490    5516    5542    5568    5595    5623    5651    5679    5707    5735    5765    5791    5826    5852    5883
          5909    5939    5964    5990    6017    6043    6068    6094    6120    6146    6172    6198    6224    6249    6275
          6301    6327    6353    6378    6404    6430    6456    6482    6508    6534    6560    6586    6614    6643    6675
          6707    6734    6761    6788    6815    6842    6869    6895    6922    6949    6976    7003    7030    7057    7084
          7111    7138    7165    7192    7219    7246    7272    7298    7324    7350    7376    7403    7425    7451    7472
          7499    7525    7551    7572    7599    7626    7650    7674    7699    7723    7748    7767    7787    7808    7829
          7858    7887    7916    7945    7974    8003    8032    8061    8083    8105    8127    8149    8175    8202    8229
          8256    8285    8314    8347    8376    8403    8430    8457    8484    8511    8540    8569    8597    8624    8651
          8678    8708    8739    8765    8791    8817    8843    8864    8885    8917    8949    8981    9013    9050    9087
          9124    9161    9198    9235    9276    9313    9345    9377    9409    9441    9462    9483    9504    9525    9547
          9568    9595    9623    9643    9662    9686    9710    9731    9758    9785    9812    9839    9861    9883    9905
          9927    9949    9971    9993   10020   10051   10078   10105   10133   10160   10187   10214   10240   10269   10298
         10327   10356   10385   10413   10442   10471   10499   10526   10553   10581   10608   10635   10661   10688   10715
         10742   10771   10800   10830   10860   10890   10924   10954   10984   11013   11032   11053   11075   11097   11119
         11141   11163   11185   11206   11228   11249   11271   11293   11315   11342   11364   11396   11422   11453   11475
         11505   11531   11557   11583   11613   11643   11671   11702   11731   11756   11785   11818   11850   11883   11911
         11945   11961   11988   12007   12026   12045   12064   12105   12142   12179   12239   12299   12323   12341   12359
         12389   12421   12454   12487   12544   12601   12632   12663   12694   12725   12756   12792   12827   12872   12955
```

```
        12975   13015   13053   13078   13103   13128   13152   13191   13229   13281   13341   13401   13431   13463   13495
        13518   13541   13563   13596   13631   13666   13701   13727   13754   13781   13811   13853   13881   13896   13921
        13937   13954   14003   14063   14123   14183   14229   14273   14303   14333   14364   14400   14436   14472   14508
        14539   14563   14592   14621   14650   14676   14718   14742   14767   14791   14819   14848   14896   14939   14990
        15034   15076   15154   15217   15284   15325   15401   15442   15567   15642
TRMTRP  16356#
TYPBIN      1#    827#
TYPDEC      1#    827#
TYPNAM      1#    827#
TYPNUM      1#    827#
TYPOCS      1#    827#
TYPOCT      1#    827#  16075   16099
TYPTXT      1#    827#
UPCODE  15722#  15747
YESCOP   1263#   4645    4659    4673    4693    4715    4735    4757    4771    4786    4801    4816    4831    4845    4860
         4875    4890    4916    4937    4963    4980    5032    5049    5075    5093    5119    5145    5197    5224    5250
         5276    5302    5329    5356    5383    5410    5436    5462    5488    5514    5540    5566    5593    5621    5649
         5677    5705    5733    5763    5789    5824    5850    5881    5907    5937    5962    5988    6015    6041    6066
         6092    6118    6144    6170    6196    6222    6247    6273    6299    6325    6351    6376    6402    6428    6454
         6480    6506    6532    6558    6584    6612    6641    6673    6705    6732    6759    6786    6813    6840    6867
         6893    6920    6947    6974    7001    7028    7055    7082    7109    7136    7163    7190    7217    7244    7270
         7296    7322    7348    7374    7401    7423    7449    7470    7497    7523    7549    7570    7597    7624    7648
         7672    7697    7721    7746    7765    7785    7806    7827    7856    7885    7914    7943    7972    8001    8030
         8059    8081    8103    8125    8147    8173    8200    8227    8254    8283    8312    8345    8374    8401    8428
         8455    8482    8509    8538    8567    8595    8622    8649    8676    8706    8737    8763    8789    8815    8841
         8862    8883    8915    8947    8979    9011    9048    9085    9122    9159    9196    9233    9274    9311    9343
         9375    9407    9439    9460    9481    9502    9523    9545    9566    9593    9621    9641    9660    9684    9708
         9729    9756    9783    9810    9837    9859    9881    9903    9925    9947    9969    9991   10018   10049   10076
        10103   10131   10158   10185   10212   10238   10267   10296   10325   10354   10383   10411   10440   10469   10497
        10524   10551   10579   10606   10633   10659   10686   10713   10740   10769   10798   10828   10858   10888   10922
        10952   10982   11011   11030   11051   11073   11095   11117   11139   11161   11183   11204   11226   11247   11269
        11291   11313   11340   11362   11394   11420   11451   11473   11503   11529   11555   11581   11611   11641   11669
        11700   11729   11754   11783   11816   11848   11881   11909   11943   11986   12005   12024   12062   12103   12140
        12177   12237   12297   12321   12339   12357   12387   12419   12452   12485   12542   12599   12630   12661   12692
        12723   12754   12790   12825   12870   12953   12973   13013   13051   13076   13101   13126   13150   13189   13227
        13279   13339   13399   13429   13461   13493   13516   13539   13561   13594   13629   13664   13879   13894   13919
        13935   13952   14674   14716   14740   14765
$$CMRE   871#    910     911     912     913     914     915
$$CMTM   871#    916     917     918     919     920
$$ESCA      1#    827#
$$NEWT      1#    827#  1936    1949    1963    1975    1989    2004    2018    2035    2051    2068    2090    2112    2130
         2148    2167    2186    2203    2226    2249    2266    2288    2314    2344    2363    2381    2397    2415    2434
         2455    2475    2496    2517    2539    2563    2586    2609    2630    2646    2665    2683    2706    2723    2738
         2756    2773    2798    2823    2842    2866    2889    2905    2923    2949    2977    2995    3013    3031    3048
         3072    3088    3106    3125    3157    3194    3231    3268    3305    3330    3355    3380    3406    3432    3458
         3484    3510    3532    3555    3591    3627    3645    3663    3681    3704    3720    3748    3762    3776    3790
         3805    3820    3835    3849    3864    3879    3894    3909    3924    3939    3953    3968    3983    3998    4013
         4027    4042    4057    4072    4098    4116    4133    4169    4190    4214    4245    4267    4307    4328    4368
         4391    4413    4432    4456    4477    4505    4526    4544    4562    4642    4656    4670    4690    4712    4732
         4754    4768    4783    4798    4813    4828    4842    4857    4872    4887    4913    4934    4960    4977    5029
         5046    5072    5090    5116    5142    5194    5221    5247    5273    5299    5326    5353    5380    5407    5433
         5459    5485    5511    5537    5563    5590    5618    5646    5674    5702    5730    5760    5786    5821    5847
         5878    5904    5934    5959    5985    6012    6038    6063    6089    6115    6141    6167    6193    6219    6244
         6270    6296    6322    6348    6373    6399    6425    6451    6477    6503    6529    6555    6581    6609    6638
         6670    6702    6729    6756    6783    6810    6837    6864    6890    6917    6944    6971    6998    7025    7052
         7079    7106    7133    7160    7187    7214    7241    7267    7293    7319    7345    7371    7398    7420    7446
```

```
        7467    7494    7520    7546    7567    7594    7621    7645    7669    7694    7718    7743    7762    7782    7803
        7824    7853    7882    7911    7940    7969    7998    8027    8056    8078    8100    8122    8144    8170    8197
        8224    8251    8280    8309    8342    8371    8398    8425    8452    8479    8506    8535    8564    8592    8619
        8646    8673    8703    8734    8760    8786    8812    8838    8859    8880    8912    8944    8976    9008    9045
        9082    9119    9156    9193    9230    9271    9308    9340    9372    9404    9436    9457    9478    9499    9520
        9542    9563    9590    9618    9638    9657    9681    9705    9726    9753    9780    9807    9834    9856    9878
        9900    9922    9944    9966    9988   10015   10046   10073   10100   10128   10155   10182   10209   10235   10264
       10293   10322   10351   10380   10408   10437   10466   10494   10521   10548   10576   10603   10630   10656   10683
       10710   10737   10766   10795   10825   10855   10885   10919   10949   10979   11008   11027   11048   11070   11092
       11114   11136   11158   11180   11201   11223   11244   11266   11288   11310   11337   11359   11391   11417   11448
       11470   11500   11526   11552   11578   11608   11638   11666   11697   11726   11751   11780   11813   11845   11878
       11906   11940   11956   11983   12002   12021   12040   12059   12100   12137   12174   12234   12294   12318   12336
       12354   12384   12416   12449   12482   12539   12596   12627   12658   12689   12720   12751   12787   12819   12863
       12950   12970   13010   13048   13073   13098   13123   13147   13186   13224   13276   13336   13396   13426   13458
       13490   13513   13536   13558   13589   13626   13661   13696   13722   13749   13776   13806   13846   13876   13891
       13916   13932   13949   13969   14029   14089   14149   14209   14254   14298   14328   14359   14395   14431   14467
       14503   14534   14558   14587   14616   14645   14667   14713   14737   14762   14786   14814   14843   14888   14926
       14977   15023   15058   15145   15206   15276   15313   15390   15429   15551   15638
$$SET  16356#  16365   16366   16367
$$SKIP     1#    827#   1944    1958    1971    1984    1999    2013    2031    2047    2063    2085    2107    2125    2143
        2162    2181    2199    2221    2244    2261    2283    2309    2338    2358    2376    2392    2410    2429    2450
        2471    2491    2512    2534    2557    2580    2603    2641    2660    2678    2701    2718    2751    2768    2793
        2818    2837    2862    2884    2901    2918    2944    2972    2990    3008    3026    3043    3083    3101    3120
        3152    3188    3225    3262    3299    3326    3351    3376    3401    3427    3453    3479    3505    3528    3550
        3641    3659    3677    3699    3715    3743    3757    3771    3785    3800    3815    3830    3844    3859    3874
        3889    3904    3919    3934    3948    3963    3978    3993    4008    4022    4037    4052    4067    4093    4111
        4128    4164    4185    4209    4240    4261    4301    4362    4408    4428    4472    4518    4538    4556    4652
        4666    4686    4708    4728    4750    4764    4779    4794    4809    4824    4838    4853    4868    4883    4909
        4930    4956    4973    5025    5042    5068    5086    5112    5138    5190    5217    5243    5269    5295    5322
        5349    5376    5403    5429    5455    5481    5507    5533    5559    5586    5613    5641    5669    5697    5725
        5755    5781    5816    5842    5873    5899    5929    5954    5981    6008    6034    6059    6085    6111    6137
        6163    6189    6215    6240    6266    6292    6318    6344    6369    6395    6421    6447    6473    6499    6525
        6551    6577    6604    6633    6665    6697    6724    6751    6778    6805    6832    6859    6885    6912    6939
        6966    6993    7020    7047    7074    7101    7128    7155    7182    7209    7236    7263    7289    7315    7341
        7367    7393    7415    7441    7462    7489    7515    7541    7562    7590    7617    7641    7665    7690    7714
        7739    7758    7778    7798    7819    7849    7878    7907    7936    7965    7994    8023    8052    8073    8095
        8117    8139    8166    8193    8220    8247    8275    8304    8337    8366    8394    8421    8448    8475    8502
        8530    8559    8587    8615    8642    8669    8698    8729    8756    8782    8808    8834    8854    8875    8906
        8938    8970    9002    9039    9076    9113    9150    9187    9224    9265    9302    9334    9366    9398    9430
        9452    9473    9494    9515    9536    9558    9586    9614    9634    9653    9677    9701    9721    9749    9775
        9802    9829    9851    9873    9895    9917    9939    9961    9983   10011   10042   10069   10096   10123   10151
       10178   10205   10231   10259   10288   10317   10346   10375   10404   10432   10461   10489   10517   10544   10572
       10599   10626   10653   10679   10706   10733   10761   10790   10820   10850   10880   10914   10944   10974   11003
       11023   11043   11065   11087   11109   11131   11153   11175   11196   11218   11239   11261   11279   11284   11301
       11306   11323   11333   11350   11355   11372   11382   11385   11404   11409   11412   11430   11434   11444   11461
       11466   11483   11493   11496   11513   11518   11521   11539   11544   11547   11565   11570   11573   11591   11594
       11599   11602   11621   11624   11629   11632   11657   11770   11799   11835   11864   11897   11926   11951   11977
       11980   11998   12017   12036   12055   12092   12129   12166   12193   12201   12213   12222   12253   12261   12273
       12282   12331   12349   13181   13219   13258   13327   13387   13612   13647   13682   14245   14289   14324   14355
       14391   14427   14463   14499   14530   14554   14583   14612   14641   14663   14706   14734   14759   14783   14811
       14840   14872
.EQUAT     1#    695#    717
.MEADE     1#    695#    696
.KT11      1#
.SETUP     1#
.SWRMI     1#    695#    705
```

```
.SWRLO    716#
.$ACT1      1#    695#    861
.$APTB      1#    695#    927#
.$APTH      1#    695#    839
.$APTY      1#    695#  16275
.$ASTA      1#
.$CATC      1#    695#    827
.$CMTA      1#    695#    871
.$DB2D      1#
.$DB2O      1#
.$DIV       1#
.$EOP       1#    695#  15681
.$ERRO      1#    695#  15998
.$ERRT      1#    695#  16061
.$MULT      1#
.$POWE      1#    695#  15722
.$RAND      1#
.$RDDE      1#
.$RDOC      1#
.$READ      1#
.$R2AZ      1#
.$SAVE      1#
.$SB2D      1#
.$SB2O      1#
.$SCOP      1#    695#  15916
.$SIZE      1#
.$SUPR      1#
.$TRAP      1#    695#  16333
.$TYPB      1#
.$TYPD      1#    695#
.$TYPE      1#    695#  16117
.$TYPO      1#    695#  16197
.$40CA      1#
.1170       1#


. ABS.  070034      000


 ERRORS DETECTED:  0

 DSKZ:CQKDAD.BIN,CQKDAD.LST/CRF/SOL/NL:TOC=CQKDAD.SML,CQKDAD.P11
 RUN-TIME:  179 227 18 SECONDS
 RUN-TIME RATIO: 762/425=1.7
 CORE USED:  39K  (77 PAGES)
```