LSI-11

4K SYSTEM EXERCISER

MD-11-DVKAH-A

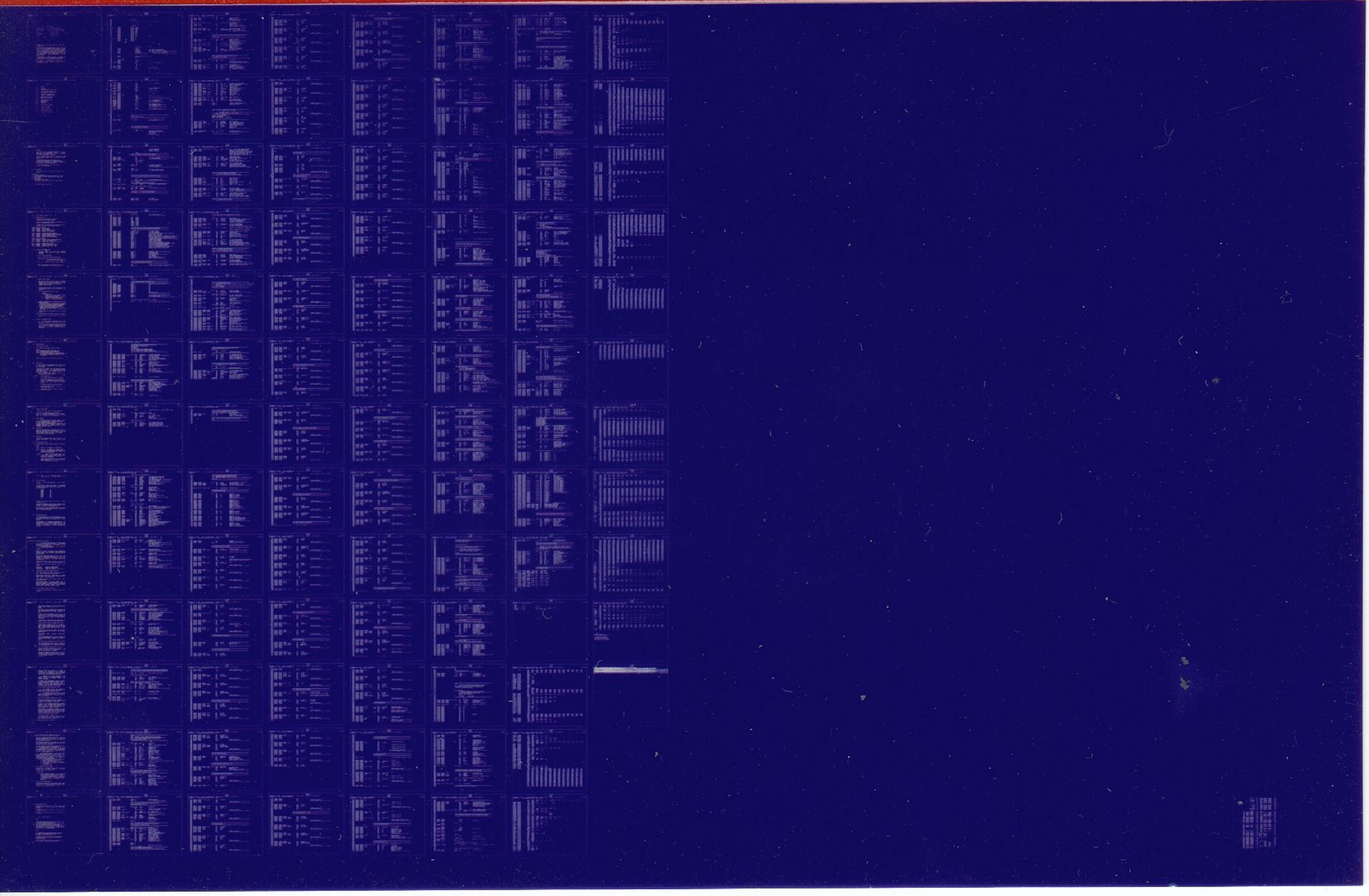
EP-DVKAH-A-DL-A

FICHE 1 OF 1

OCT 1976

digital

Made in U.S.A.



```
.REPT 0

                        IDENTIFICATION

        PRODUCT CODE:           MAINDEC-11-DVKAH-A-D
        PRODUCT NAME:           LSI-11 4K SYSTEM EXERCISER
        DATE :                  MARCH 21, 1976
        MAINTAINER:             DIAGNOSTIC GROUP
        AUTHOR:                 FRED STRAIGHT
```

CONTENTS

[1.0] ABSTRACT

THIS IS AN LSI-11 4K SYSTEMS EXERCISER. IT IS A TEST OF THE PROCESSORS ABILITY TO OPERATE PERIPHERALS IN INTERRUPT MODE. IT IS NOT A COMPLETE TEST OF THE PERIPHERALS THEMSELVES.

THE TEST OCCUPIES LESS THAN 4K OF MEMORY BUT DOES RUN A MEMORY ADDRESS TEST OF ALL AVAILABLE MEMORY ABOVE 4K.

A PROCESSOR INSTRUCTION TEST IS RUN WHILE ALLOWING THE PERIPHERALS TO INTERRUPT AT RANDOM. IF RELOCATION IS ENABLED THE PROCESSOR INSTRUCTION TEST IS RUN IN EACH 4K MEMORY BANK.

IT WILL RUN UNDER ACT-11 OR RXDP MONITORS.
IT IS NOT APT COMPATIBLE.

[2.0] EQUIPMENT

LSI-11 WITH MINIMUM 4K OF MEMORY AND CONSOLE OUTPUT DEVICE.

OPTIONAL:
A. UP TO 28K OF MEMORY
B. TELETYPE ASR33 LOW SPEED READER (REQUIRES TEST TAPE)
C. DRV11-PARALLEL LINE UNIT (REQUIRES TEST CABLE BC08R)
E. LINE CLOCK
F. LINE PRINTER
G. RXV11-FLOPPY DISK SYSTEM
    (REQUIRES A SCRATCH DISKETTE FOR EACH DRIVE TESTED)

[2.1] PROGRAM STORAGE = 0000-17476

[3.0] START/RESTART ADDRESS = 200

[3.1] SWITCH SETTINGS

SOFTWARE SWITCH REGISTER = LOCATION 176
LOCATION 176= 114177 WHEN LOADED.

"STATIC" SWITCH SETTINGS CAN ONLY BE MODIFIED IF THE TEST IS TO BE RESTARTED AT 200.

"DYNAMIC" SWITCH SETTINGS CAN BE MODIFIED DURING A CONTROL-C [↑C] HALT AND THE TEST CONTINUED.

[DYNAMIC SWITCH SETTINGS]
BIT15 (100000) - HALT ON ERROR
BIT14 (040000) - LOOP ON SUBTEST
BIT13 (020000) - INHIBIT ERROR PRINTOUTS
BIT12 (010000) - INHIBIT T BIT TRAPPING

BIT11 (004000) - INHIBIT SUB TEST ITERATION
BIT10 (002000) - INHIBIT PROCESSOR INSTRUCTION TEST
BIT 9 (001000) - INHIBIT INSTRUCTION TEST RELOCATION
BIT 8 (000400) - RESTART PROGRAM ON ERROR.
BIT 7 (000200) - INHIBIT END OF PASS PRINTOUT

[STATIC SWITCH SETTINGS]
BIT 6 (000100) - INHIBIT ASR33 LOW SPEED READER TEST
BIT 5 (000040) - INHIBIT LINE PRINTER TEST
BIT 4 (000020) - INHIBIT DRV11 PARALLEL LINE UNIT TEST

BIT 3 (000010) - INHIBIT EIS/FIS TEST
BIT 2 (000004) - INHIBIT CONSOLE OUTPUT TEST
BIT 1 (000002) - INHIBIT FLOPPY UNIT 1 TEST
BIT 0 (000001) - INHIBIT FLOPPY UNIT 0 TEST

[4.0] OPERATOR ACTION REQUIRED

A. LOAD THE PROGRAM INTO MEMORY USING STANDARD PROCEDURE FOR PDP-11 BINARY FORMATTED PROGRAMS.

B. IF I/O DEVICES SELECTED:

FLOPPY- INSURE SCRATCH DISKETTES INSTALLED

LINEPRINTER- IF 132 COLUMN PRINTER THEN CHANGE LOCATION LP80 [542] FROM 117 TO 203

DRV11 (PLU)- INSURE TEST CABLE (BC08R) INSTALLED

ASR33- PLACE BINARY TEST TAPE (SEE SEC. 8.0) IN THE LOW SPEED READER. THE TEST WILL ALLOW ANY

NUMBER OF BLANK FRAMES.

C. SET SOFTWARE SWITCH [LOCATION 176] TO DESIRED OPTIONS. IF AN OPTION IS SELECTED AND ITS ADDRESS FAILS TO ANSWER THE TEST WILL HALT AT PC = 0602. NOTE: LOC 176 =114177 WHEN THE PROGRAM IS FIRST LOADED.

D. START THE TEST AT 200. IF THE TEST HALTS AT PC = 602 RECHECK SWITCH OPTIONS IN LOCATION 176 AND RESTART.

E. THE FOLLOWING PRINTOUTS WILL OCCUR:

"DVKAH-A"
"MEMORY= XXXXXX"
WHERE "XXXX" IS THE HIGHEST MEMORY LOCATION FOUND BY SIZING.
NOTE: IF RELOCATION INHIBITED OR RUNNING IN MEMORY WILL NOT BE SIZED AND "MEMORY = 17776" WILL BE PRINTED.

F. CORRECT OPERATION FOR THE LINEPRINTER AND CONSOLE TERMINAL IS VISUAL:
1. LINEPRINTER OUTPUT IS AN INCREMENTING CHARACTER PATTERN THAT INCREMENTS ACROSS LINE BOUNDARIES SO THAT EACH SUCCESSIVE LINE IS DIFFERENT.
2. CONSOLE OUTPUT IS AN INCREMENTING CHARACTER PATTERN THAT REPEATS EVERY LINE. IN ADDITION 15 NULLS ARE PRINTED AT THE BEGINNING OF EACH LINE.

G. END OF PASS WILL BE INDICATED BY THE FOLLOWING PRINTOUT.

"PASS= 00001 ERROR= 00000 RXERROR= 00000 TIME= 00000"

FOR DEFINITION OF ABOVE PRINTOUT SEE SECTION 7.0.

[4.1] CONTROL-C INTERRUPT OPTION

THE TEST CAN BE INTERRUPTED AT ANY TIME BY TYPING CONTROL-C. THE TEST WILL HALT AT PC=172.

THE SWITCH OPTIONS [LOCATION 176] CAN BE MODIFIED AND THE PROGRAM CONTINUED BY TYPING "P". NOTE: IF THE ASR 33 LOW SPEED READER IS UNDER TEST THE CONTROL-C OPTION WILL CAUSE A DATA ERROR.

[5.0] ERROR PRINTOUTS

ERROR PRINTOUT FORMAT:

"PC = 000000 PSW = 000000 RETURN = 000000"

WHERE:
PC = THE PROGRAM COUNTER WHERE ERROR FOUND.
PSW = PROCESSOR STATUS AT TIME OF ERROR
RETURN = THE BEGINNING ADDRESS OF THE INSTRUCTION
SUBTEST BEING EXECUTED.THE "RETURN" ADDRESS CAN
BE USED AS THE BEGINNING ADDRESS FOR A LOOP
ON SUBTEST.

REFER TO LISTING AT FAILING LOCATION FOR DETAILS.

[5.1] ERROR HALTS

IF A HALT OCCURS WITH NO ERROR PRINTOUT, FIRST LOOK AT THE LISTING TO DETERMINE IF THE PC POINTS TO A HALT INSTRUCTION.

IF THE FAILING ADDRESS DOES NOT CONTAIN A HALT INSTRUCTION THE LSI-11 "M" COMMAND CAN DETERMINE WHAT CAUSED THE HALT. FOR REFERENCE HERE IS AN EXPLANATION OF THE OCTAL DIGITS PRINTED VIA THE LSI-11 "M" COMMAND.
NOTE: 6 CHARACTERS ARE PRINTED BUT ONLY THE RIGHTMOST OCTAL DIGIT IS USED.

!DIGIT!	!WHAT CAUSED HALT!
0	HALT INSTRUCTION OR BREAK KEY ON TERMINAL.
1	BUS ERROR OCCURRED WHILE GETTING INTERRUPT VECTOR. IT'S POSSIBLE THAT THE OPTION IS IN A SLOT THAT IS NOT WIRED FOR THE IACK SIGNAL.
2	BUS ERROR OCCURRED WHILE DOING REFRESH.
3	DOUBLE BUS ERROR OCCURRED. (STACK (R6) WAS NON-EXISTENT VALUE).
4	NON-EXISTENT MICRO-PC ADDRESS OCCURRED ON INTERNAL CPU BUS.

[6.0] EXECUTION TIME TABLE

TYPICAL TEST TIMES: (IN FORM - MINUTES:SECONDS)

7:30 = 4K + ALL OPTIONS + ALL I/O DEVICES.
5:25 = 4K + NO TBIT + ITERATIONS + ALL I/O DEVICES
1:25 = 4K + NO TBIT + NO ITERATIONS + ALL I/O DEVICES.
0:50 = 4K + NO OPTIONS + NO I/O DEVICES.

[6.1] STACK POINTER

THE STACK STARTS AT LOCATION "STACK", WHICH STARTS AT LOCATION 17474. IT IS NEVER RELOCATED. IF STACK PUSHDOWN OVERFLOW OCCURS IT WILL DESTROY THE MESSAGE STORAGE AREA FIRST. IF MESSAGES ARE SUDDENLY GARBLED, SUSPECT STACK PUSHDOWN OVERFLOW. THIS COULD HAPPEN IF A DEVICE'S INTERRUPT CIRCUITRY FAILED.

[6.2] PASS COUNT

A PASS IS DEFINED AS THE INSTRUCTION TEST MAKING A PASS IN EACH SELECTED 4K MEMORY BANK. IF T BIT TRAPPING IS SELECTED (BIT12=0) THEN ANOTHER TEST PASS IN BANK 0 IS DONE WITH THE T BIT ON BEFORE LOCATION "PASS" [524] IS UPDATED.

[6.3] POWER FAIL

THE TEST WILL RECOVER FROM A POWER FAILURE IF IT OCCURS IN CORE MEMORY. "PWR" WILL BE TYPED ON POWER UP.

[7.0] PASS PRINTOUT FORMAT

"PASS= 000000 ERROR= 000000 RXERROR= 000000 TIME= 000000"

WHERE:

PASS = TOTAL NO. OF PASSES LOC. "PASS" [524]

ERROR = TOTAL NO. OF ERRORS. LOC. "ERROR" [526]
!THIS IS A RUNNING TOTAL OF ALL ERRORS. IT IS USEFUL TO GET AN ERROR COUNT WHEN ERROR PRINTOUTS ARE INHIBITED.

RXERROR = FLOPPY TOTAL SOFT ERROR COUNT. UNIT 1 AND UNIT 0. LOC. "RXSOFT" [516]. THE FLOPPY TEST IS RETRIED 10 TIMES BEFORE IT IS CONSIDERED A HARD ERROR.

WHILE THE SOFT ERROR COUNT GIVES NO DETAILED ERROR DESCRIPTION, IF IT IS NOT 0000 THE FLOPPY DIAGNOSTIC SHOULD BE RUN.

TIME = TOTAL NO. OF INTERRUPTS FROM THE LINE CLOCK, IN OCTAL. LOC. "TIME" [3574]

[8.0] UTILITY ROUTINES

UTILITY #1 - PRINT ERROR REGISTERS - (START LOCATION = 410)

THIS ROUTINE WILL PRINT THE VALUE OF ALL REGISTERS AT THE TIME OF THE LAST ERROR. THE LOCATIONS CONTAINING THE REGISTER VALUES ARE PRINTED IN THE FOLLOWING ORDER:

```
SAVR0           ;R0
SAVR1           ;R1
SAVR2           ;R2
SAVR3           ;R3
SAVR4           ;R4
SAVR5           ;R5
SAVSP           ;R6
SAVPC           ;R7
SAVPS           ;PSW
```

UTILITY #2 - PUNCH BINARY TEST TAPE - (START LOCATION = 400)

THIS ROUTINE PUNCHES A TEST TAPE IN A BINARY COUNT PATTERN. THE TAPE CAN THEN BE USED AS AN INPUT TEST TAPE FOR THE ASR33 LOW SPEED READER TEST.

THE ROUTINE WILL PUNCH A BINARY PATTERN FROM 0-377 THEN 15 LINES OF BLANK TAPE, REPEATING THIS SEQUENCE UNTIL THE MACHINE IS HALTED.

[9.0] PROGRAM DESCRIPTION

THIS TEST CONSISTS OF A BASIC INSTRUCTION TEST WITH I/O INTERRUPT ROUTINES. THE I/O INTERRUPT ROUTINES ARE ACTUALLY INDIVIDUAL TEST ROUTINES FOR A GIVEN DEVICE.

THE FOLLOWING IS A DETAILED DESCRIPTION OF THE PROGRAM AND THE I/O INTERRUPT SERVICE ROUTINES. THE DESCRIPTION IS GIVEN IN THE SAME ORDER AS THE PROGRAM EXISTS IN MEMORY. THE NAME OF THE ROUTINE IS GIVEN IN BRACKETS TO AID IN FOLLOWING THE LISTING.

[9.1] TRAPCATCHER [LOC 0-776]

ALL UNUSED LOCATIONS FROM 4 TO 776 ARE SETUP TO HALT IF AN ILLEGAL TRAP OCCURS. EACH POSSIBLE TRAP ADDRESS IS LOADED WITH ITS ADDRESS +2, AND ADDRESS +2 CONTAINS A HALT.
LOCATION 00000 CONTAINS A HALT INSTRUCTION TO CATCH ILLEGAL VECTOR TRAPS.

EXAMPLE:

ASSUME AN ILLEGAL INSTRUCTION TRAP OCCURRED AT LOCATION 1000. THE FOLLOWING DESCRIBES PROGRAM CONTROL AND HOW TO FIND THAT THE TRAP OCCURRED AT LOCATION 1000.

THE ILLEGAL INSTRUCTION CAUSES THE CPU TO PUSH THE CURRENT PC (1002) AND STATUS ONTO THE STACK. THEN THE NEW PC IS PICKED UP AT LOCATION 10. LOCATION 10 CONTAINS A 12 (ADDRESS +2), SO PROGRAM CONTROL GOES TO LOCATION 12.

LOCATION 12 CONTAINS A HALT, SO THE CPU HALTS AND PRINTS PC = 14. AT THIS TIME USING MICRO-ODT THE

LOCATION WHERE THE TRAP OCCURRED (1000) CAN BE FOUND.

```
@R6/17052       !FINDS THE STACK ADDRESS
@17052/1000     !FINDS THE TRAPPED PC
@17054/1000     !FINDS CPU STATUS AT TIME OF TRAP
```

[9.2] INITIALIZATION AND DEVICE SETUP [START]

SETUP STACK POINTER. IF FIRST TIME CLEAR PASS COUNT ETC. CHECK FOR HARDWARE SWITCH REGISTER.

SETUP CONSOLE OUTPUT TEST. SETUP ASR 33 LOW SPEED READER TEST. SETUP HIGH SPEED READER AND PUNCH TEST. SETUP RXV11 FLOPPY TEST.

SETUP DRV11 PLU TEST. SETUP EIS/FIS TEST. SETUP LINE PRINTER TEST.

SETUP RELOCATION AND RUN MEMORY ADDRESS TEST IF RELOCATION SELECTED. PRINT "MEMORY =". SETUP LOC PASSNO TO NUMBER OF INSTRUCTION TEST PASSES REQUIRED BEFORE "PASS" PRINTED. PASSNO = 2 IF T BIT TRAP OR RELOCATION TEST SELECTED ELSE = 00000.

[9.3] DEVICE INTERRUPT SERVICE ROUTINES

A. ASR 33 LOW SPEED READER INTERRUPT SERVICE/TEST [ASRINR]

THIS ROUTINE ASSUMES A TEST TAPE CONSISTING OF A BINARY COUNT PATTERN FROM 0-377 IS IN THE READER.

THE TEST WILL READ A CHARACTER PER INTERRUPT AND WILL ACCEPT ANY NUMBER OF BLANKS. THE FIRST NON ZERO BYTE MUST BE A 1 AND THEN SEQUENTIAL TO 377.

A TEST TAPE CAN BE GENERATED USING UTILITY #2 SECTION 8.0 OR CAN BE PRODUCED BY RUNNING THE CONSOLE OUTPUT TEST WITH THE PUNCH ENABLED. NOTE IF THE PUNCH IS USED THE TYPEOUTS MUST BE SUPPRESSED.

IF THE TEST IS RESTARTED THE TAPE MUST BE RESTARTED SINCE THERE IS NO RESYNCHRONIZATION.

B. CONSOLE OUTPUT INTERRUPT SERVICE/TEST [TYOUTR]

THIS ROUTINE WILL PUNCH A TAPE OR PRINTOUT A BINARY COUNT PATTERN FROM 0-377 IN INTERRUPT MODE. FIRST 15 BLANK LINES ARE OUTPUT SO THE TAPE CAN BE ALIGNED WHEN RUNNING THE LOW SPEED READER TEST.

TO CHECK THE VALIDITY OF THE DATA, THE TEST TAPE SHOULD BE USED AS AN INPUT TAPE TO THE LOW SPEED READER TEST.

C. HIGH SPEED READER INTERRUPT SERVICE/TEST [HSRINR]

THIS IS FOR MANUFACTURING TO ALLOW TESTING OF THE HIGH SPEED READER. THE TESTING IS IDENTICAL TO THE ASR 33 LOW SPEED READER. SEE STEP A. ABOVE.

D. HIGH SPEED PUNCH INTERRUPT SERVICE/TEST [HSPINR]

THIS IS FOR MANUFACTURING TO ALLOW TESTING OF THE HIGH SPEED PUNCH. THE TESTING IS IDENTICAL TO THE CONSOLE OUTPUT TEST. SEE STEP B. ABOVE.

E. DRV11 PLU INTERRUPT SERVICE/TEST [PLUB,PLUA]

THE TEST REQUIRES A TEST CABLE TO TIE CSR 1 TO REQUEST "B" AND CSR 0 TO REQUEST "A". IT ALSO TIES THE OUTPUT DATA TO THE INPUT.

REQUEST B IS ASSERTED IN DEVICE SETUP AND THE INTERRUPT SERVICE ROUTINE "PLUB" IS CALLED. PLUB COMPARES THE DATA TRANSMITTED AND RECEIVED, IT THEN ENABLES REQUEST "A". PLUB IS ONLY CALLED ONCE.

PLUA IS ENTERED WHEN REQUEST "A" CAUSES AN INTERRUPT. THE TRANSMITTED DATA (252525) IS COMPARED WITH THE RECEIVED. THE DATA PATTERN [LOC PLUDAT] IS INCREMENTED AND THE ROUTINE RETURNS TO THE PROCESSOR INSTRUCTION TEST.

REQUEST "A" IS THEN ASSERTED IN THE "SCOPE" ROUTINE TO REGENERATE INTERRUPTS. THE "SCOPE" ROUTINE IS ENTERED AFTER EACH INSTRUCTION SUBTEST.

TO INSURE THE PLU INTERRUPTED, LOCATION "CHECK" IS SET IN THE PLUA SERVICE ROUTINE. "CHECK" IS TESTED BEFORE END OF PASS AN ERROR HALT OCCURS IF THE PLU WAS SELECTED BUT FAILED TO INTERRUPT. THE MOST LIKELY CAUSE OF THIS ERROR IS THE FAILURE TO INSTALL THE TEST CABLE.

F. LINE PRINTER INTERRUPT SERVICE [LPINTR]

THIS ROUTINE OUTPUTS AN INCREMENTING BINARY COUNT PATTERN TO THE LINE PRINTER. IT INTERRUPTS FOR EVERY CHARACTER OUTPUT.

THE CHARACTER COUNT CAN BE SET FOR AN 80 COLUMN OR 132 COLUMN LINE PRINTER. SET LOCATION LP80 = 117 FOR 80 COLUMN. SET LOCATION LP80 = 203 FOR 132 COLUMN.

G. RXV11 FLOPPY INTERRUPT SERVICE/TEST [RXINT]

THE FLOPPY TESTING POSES A UNIQUE PROBLEM. IT IS NOT AN NPR DEVICE AND IN ADDITION MOST TRANSFERS TO IT ARE DONE IN FLAG MODE.

TO TEST IT IN AN INTERRUPT DRIVEN MODE REQUIRED SERVICING THE INTERRUPTS AND THEN KEEPING CONTROL. TO DO THE FLAG MODE TRANSFERS. IT ALSO REQUIRED A SERVICE ROUTINE DISPATCHER DEPENDING ON WHICH INTERRUPT OCCURRED.

THE NET RESULT IS A NUMBER OF SERVICE ROUTINES WHICH ARE TOO INVOLVED TO DISCUSS HERE. FOR A DETAILED DESCRIPTION REFER TO THE FOLLOWING ROUTINES IN THE LISTING: RXINT, RXINER, RXFRST, RXWSEC, RXRSEC, RXRD, RXWTTR, RXWTDN.

H. LINE CLOCK INTERRUPT SERVICE [CLOCK]

THIS ROUTINE SIMPLY INCREMENTS LOCATION "TIME" AND RETURNS VIA RTI. THE TIME IS PRINTED IN OCTAL IN THE END OF PASS PRINTOUT.

[9.4] PROCESSOR INSTRUCTION TEST [BEGIN]

THIS IS A BASIC LSI-11 INSTRUCTION TEST THAT IS RUN WHILE THE I/O DEVICES INTERRUPT AT RANDOM.

ALL LSI-11 INSTRUCTIONS ARE TESTED IN AT LEAST ONE MODE FOR CORRECT OPERATION. THIS IS NOT INTENDED TO REPLACE THE BASIC LSI-11 INSTRUCTION TEST, AND THAT TEST SHOULD ALWAYS BE RUN.

IF RELOCATION IS ENABLED THIS SUBTEST WILL BE RUN IN EACH 4K MEMORY BANK. IT IS RELOCATED TO EACH MEMORY BANK WHEN THE TEST IS STARTED.

LOCATION "LOCATE" CONTAINS THE BEGINNING ADDRESS OF THE INSTRUCTION TEST WHEN IT IS RELOCATED.
SINCE THE TEST IS RELOCATED TO ITS SAME RELATIVE ADDRESS ANOTHER WAY IS TO ADD THE BANK BITS TO ADDRESS "BEGIN". FOR EXAMPLE IF THE INSTRUCTION TEST IS AT 3576 IN BANK 0 IT WILL RELOCATE TO 23576 IN BANK 1, 43576 IN BANK 2 ETC.

THE INSTRUCTION TEST IS IDENTICAL IN EACH BANK EXCEPT FOR LOCATION "TRPA". IF IN BANK 0 TRPA = "JMP @# BANK0" IF IN BANK 1 TRPA = "JMP @#BEGIN + 40000" ETC... UNTIL THE LAST MEMORY BANK WHERE TRPA = "JMP @# BANK0". THIS CAUSES THE END OF PASS ROUTINE TO BE ENTERED..
AN EXAMPLE OF THE INSTRUCTION TEST FLOW FOR 8K IS:

1. EXECUTE INSTRUCTION TEST IN BANK 0
2. EXECUTE THE TEST IN BANK 0
(IF T BIT SELECTED IT WILL BE ON THIS PASS)
3. JMP TO BANK 1 AND EXECUTE THE INSTRUCTION TEST THERE.
4. PRINT END OF PASS

[9.5] END OF SUBTEST CONTROL [SCOPEC]

THIS ROUTINE IS ENTERED AFTER EACH INSTRUCTION SUBTEST VIA TRAP CALL "SCOPE". IT CONTROLS ITERATION AND PLU INTERRUPT RATE.

ROUTINE FLOW:

IF SR BIT 14=1 THEN SET "RETURN" TO BEGINNING OF CURRENT SUBTEST (SCOPE LOOP). IF BIT 11=0 THEN UPDATE ITERATION COUNT AND RETURN TO BEGINNING OF CURRENT SUBTEST. (ITERATION). IN EITHER CASE IF PLU UNDER TEST (BIT04) THEN SET PLU REQUEST "A" TO GENERATE A PLU INTERRUPT.

[9.6] END OF TEST CONTROL [BANK0]

THIS ROUTINE DETERMINES IF THE PLU AND FLOPPY HAVE MADE A PASS. AND IF SO THEN PRINTS END OF PASS STATEMENT. IT IS ENTERED AFTER THE INSTRUCTION TEST HAS MADE A PASS IN EACH 4K BANK.

[10.0] MANUFACTURING NOTES

INCLUDED IS AN OPTIONAL TEST OF THE HIGH SPEED READER AND PUNCH, WHICH MAY BE USED IN MANUFACTURING CHECKOUT.

TO ENABLE THE HIGH SPEED READER TEST, SET HSRINH [LOC 544] TO 0.

TO ENABLE THE HIGH SPEED PUNCH TEST, SET HSPINH [LOC 546] TO 0.

[10.1] ACT11 & RXDP OPERATION

RXDP

IF IN RXDP "CHAIN" MODE THEN NO AUTO-SIZING IS DONE TO PROTECT THE RXDP CHAIN MONITOR.
SINCE THE SOFTWARE SR IS SET TO 114177 WHEN LOADED ONLY THE PROCESSOR INSTRUCTION TEST IS RUN.
ALSO WHENEVER THE TEST IS LOADED UNDER RXDP; LOC 41 [HIGH BYTE LOC 40] IS SET = 10. THIS PROTECTS THE DISKETTE USED TO LOAD THE PROGRAM. TO TEST UNITO SET LOCATION 41 = 0 BEFORE STARTING.

ACT11

IF LOADED BY ACT11 THEN AUTO-SIZING IS ENABLED AND SINCE THE SOFTWARE SR=114177 THEN ONLY THE PROCESSOR INSTRUCTION TEST IS RUN.

IF LOADED BY ACT11 "DUMP" MODE, THE USER SHOULD MODIFY LOCATION 176 TO SELECT OPTIONS BEFORE STARTING THE TEST.

ALSO IF IN AUTO MODE UNDER ACT11 (42=$ENDAD) THE TITLE PRINTOUT ("DVKAH") IS INHIBITED.

.ENDR

```
                                .ENABL AMA
                                .TITLE  MAINDEC-11-DVKAH-A     LSI-11 4K SYSTEM TEST


                                ;*EQUATES

            100000              BIT15=  100000
            040000              BIT14=  40000
            020000              BIT13=  20000
            010000              BIT12=  10000
            004000              BIT11=  4000
            002000              BIT10=  2000
            001000              BIT09=  1000
            000400              BIT08=  400
            000200              BIT07=  200
            000100              BIT06=  100
            000040              BIT05=  40
            000020              BIT04=  20
            000010              BIT03=  10
            000004              BIT02=  4
            000002              BIT01=  2
            000001              BIT00=  1


                                .ENABLE ABS

            000240              NOP=240
            104000              HLT=EMT                     ;TRAP USED FOR ERROR PRINTOUT
            104001              TYPE=EMT+1                  ;TRAP USED FOR MESSAGE PRINTOUTS
            104400              SCOPE=TRAP                  ;TRAP USED SCOPE LOOP AND ITERATION OF SUB PROBLEMS
            000004              PNTCHR=IOT                  ;IOT TRAP USED TO PRINT A CHARACTER
            000023              CHECKM=BIT04+BIT01+BIT00        ;SR SETTINGS OF DEVICES CHECKED FOR PASSES


            000230              CWAIT=  BIT07+BIT04+BIT03
            000176              SR=SWREG


            000000              XX=0
            000000              XXXXX=0
            000000              .=0
000000      000000              HALT                        ;TO CATCH ILLEGAL TRAPS.
            000004              .=4
                                .LIST   ME
            000014              .=14
000014      000016              .+2
000016      000000              HALT                        ;FALSE TRACE TRAP
            000020              .=20
000020      016216              OUTCHR                      ;PRINT A CHAR. ROUTINE
```

```
                                                340                     ;LOCK OUT INTERRUPTS
                                                .=24
                                                PFAIL
                                                340
                                                .=30
                                                HLTMES                  ;FOR HLT/MESSAGE TRAPS
                                                340                     ;HIGHEST PRIORITY
                                                .=34
                                                $SCOPEC                 ;SCOPE LOOP TRAP
                                                340

                                                0                       ;ACT11 HOOKS
                                                0
                                                .=46
                                                $ENDAD                  ;POINT ACT11 TO EOP LOC.
                                                .=52
                                                .WORD 40000             ;TELL ACT11 RUNTIME IS MEMORY DEPENDENT
                                                .=60
                                                ASRINR                  ;SLU IN INTERRUPT VECTOR
                                                340                     ;LOCK OUT INTERRUPTS
                                                TYOUTR                  ;SLU OUT INTERRUPT VECTOR
                                                0
                                                HSRINR                  ;HSR INTERRUPT VECTOR
                                                0                       ;ALLOW INTERRUPTS IN HSR ROUTINE
                                                HPOUTR                  ;HSP INTERRUPT VECTOR
                                                0                       ;ALLOW INTERRUPTS IN HSP ROUTINE
                                                .=100
                                                CLOCK                   ;CLOCK INTERRUPT VECTOR
                                                340                     ;STOP INTERRUPTS IN CLOCK ROUTINE


                                ;*******************************************************************************
                        000170  .=170
                                .SBTTL CONTROL-C HALT
                                ;*******************************************************************************

000170  000000          CTCHLT: HALT                    ;CNTRL-C PRESSED ON SLU INPUT
000172  000002                  RTI                     ;IF PROCEED RETURN TO TEST


                                ;*******************************************************************************
                                .SBTTL SOFTWARE SWITCH REGISTER
                                ;*******************************************************************************

                        000176          .=176
000176  114177                  SWREG:  114177          ;SOFTWARE SWITCH REGISTER
                                                        ;DEFAULTS TO PROCESSOR TEST,
                                                        ;HALT ON ERROR
                                                        ;NO EIS/FIS
                                                        ;NO OPTIONS SELECTED.
```

```
                                                        ;T BIT NOT SELECTED
                                                        ;ENABLE RELOCATION
                                                        ;INHIBIT ITERATIONS

                000200                          .=200
                                ;*******************************************************************
                                .SBTTL LP INTERRUPT VECTOR AND START ADDRESS
                                ;*******************************************************************

000200  000402                          BR      .+6             ;THIS IS LP INTERRUPT VECTOR AND START ADDRESS
000202  000340                          340                     ;LP INTERRUPT PRIORITY
000204  000240                          NOP                     ;THIS IS SKIPPED ON START AT 200
000206  000137  000602                  JMP     @#START
                                        .NLIST  MC
                                        .LIST   MD,ME

                000244                  .=244
000244  000246                  FISVEC: .+2                     ;FIS ABORT TRAP ENTRANCE
000246  104000                  FISLVL: HLT                     ;*ERROR* FIS ABORTED
                                                                ;THE LOOP ADDRESS PRINTED TELLS WHAT FIS ROUTINE ABORTED

                000264          .=264
000264  002606                  RXVEC:  RXINT                   ;RX FLOPPY INTERRUPT VECTOR
000266  000340                  RXLVL:  340                     ;RX INTERRUPT VECTOR PRIORITY


                                ;*******************************************************************
                                .SBTTL ENTRY AT 400 FOR ASR33 TEST TAPE PUNCH ROUTINE
                                ;*******************************************************************

                000400                  .=400
000400  000405                  BR      PASR                    ;GO PUNCH ASR33 TEST TAPE.
                                ;*******************************************************************
                000402                  .=402
                                .SBTTL LP INTERRUPT VECTOR ROUTINE DISPATCH
                                ;*******************************************************************
000402  000137  002424                  JMP     LPINTR          ;GOTO ACTUAL LP SERVICE ROUTINE

                                ;START AT 410 TO PRINT ERROR REGISTERS
                000410                  .=410
000410  000137  016762          PREG:   JMP     @#PNTABL        ;PRINT REGISTERS AT LAST ERROR.
000414  000137  017030          PASR:   JMP     @#ASRUTL

                                ;*******************************************************************
                                .SBTTL BEGIN    I/O DEVICE STATUS ADDRESSES
                                ;*******************************************************************


000420  167770                  DRCSR:  167770                  ;PLU STATUS
000422  167772                  DROBUF: 167772                  ;PLU OUTPUT BUFFER
000424  167774                  DRIBUF: 167774                  ;PLU INPUT BUFFER
```

```
000426  000300                  DRVECA: 300                     ;PLU INTERRUPT VECTOR "A"
                                                                ;VECTOR "B" ASSUMED AT DRVECA+4.

000430  177560                  TKS:    177560
000432  177562                  TKB:    177562
000434  177564                  TPS:    177564
000436  177566                  TPB:    177566
000440  177550                  HRCSR:  177550
000442  177552                  HRDBR:  177552
000444  177554                  HPCSR:  177554
000446  177556                  HPDBR:  177556
000450  177514                  LPCSR:  177514
000452  177516                  LPDBR:  177516

                                ;*****************************************************************
                                .SBTTL RX FLOPPY DISK STATUS REGISTERS AND CONSTANTS
                                ;*****************************************************************

000454  177170                  RXCSR:  177170                  ;RX FLOPPY CSR REG.
000456  177172                  RXDB:   177172                  ;RX FLOPPY DATA REG.
000460  000264                  RXVECP: RXVEC                   ;RX FLOPPY INTERRUPT VECTOR (LEVEL ASSUMED 340)
000462  000000                  RXUN0:  0                       ;SELECT UNIT0 (LOW BYTE)
000464  000000                  RXUN1:  0                       ;SELECT UNIT1 (HIGH BYTE)
000466  000001                  RXSAB:  1                       ;BEGIN RX  SECTOR TEST NO. (VARIES 1-31 BY INCR. OF 3)
000470  000001                  RXTAB:  1                       ;BEGIN RX TRACK TEST NO. (ALWAYS ENDS AT TRACK:114)
000472  000000                  RXDAT:  0                       ;RX FLOPPY WRITE DATA BEGIN (VARIES FROM 0-177)
000474  000000                  RXUTT:  0                       ;CURRENT DRIVE SELECTED. (IF BIT 4=1 DRIVE 1 SELECTED)
000476  000000                  RXSA:   0                       ;CURRENT SECTOR ADDRESS
000500  000000                  RXTA:   0                       ;CURRENT TRACK ADDRESS
000502  000000                  RXLCNT: 0                       ;DELAY COUNT RX FLAG WAITS
000504  000000                  RXHCNT: 0                       ;DELAY COUNT RX FLAG WAITS (HIGH WORD)
000506  000000                  SRXES:  0                       ;VALUE OF RX ERROR REGISTER WHEN ERROR OCCURRED.
000510  000000                  SRXCSR: 0                       ;VALUE OF RX CSR WHEN ERROR OCCURRED.
000512  000000                  SRXDB:  0                       ;VALUE OF RXDB WHEN ERROR OCCURRED
000514  000000                  SRXSB:  0                       ;VALUE OF RX EXTENDED ERROR STATUS WHEN ERROR.
000516  000000                  RXSOFT: 0                       ;TOTAL RX SOFT ERRORS(CRC,PARITY,SEEK)
000520  000000                  RXRTRY: 0                       ;RX RETRY COUNT. (IF >:11. THEN HALT)


000522  000000                  LOCATE: 0                       ;ADDRESS OF "BEGIN" EVEN WHEN RELOCATED
000524  000000                  PASS:   0                       ;HOLDS PASS COUNT
000526  000000                  ERROR:  0                       ;HOLDS ERROR COUNT
000530  000000                  CHECK:  0                       ;BITS SET AS EACH DEVICE INTERRUPTS.
000532  000000                  RETURN: 0                       ;LOOP ADDRESS
000534  000000                  PLUDAT: 0                       ;PLU COMPARE DATA
000536  000000                  OLDR5:  0                       ;SAVES "OLDR5" FOR MARK TITEST
000540  000000                  CHAIN:  0                       ;SET IF IN RXDP CHAIN MODE


                                ;*****************************************************************
                                .SBTTL LOCATIONS MODIFIED BY OPERATOR
                                ;*****************************************************************

000542  000117                  LP80:   117                     ;CHANGE FROM 117 TO 203 IF 132 COLUMN PRINTER.
000544  177777                  HSRINH: -1                      ;IF NOT = 0 THEN INHIBIT HSR TESTING
```

```
000546  177777          HSPINH: -1                      ;IF NOT = 0 THEN INHIBIT HSP TESTING

                        ;*****************************************************************
                        .SBTTL LOCATIONS PRINTED VIA START AT 130
                        ;*****************************************************************
000550                  EFIRST:                         ;DEFINES FIRST LOCATION PRINTED AT ST 130
000550  000000          SAVR0:  0                       ;R0
000552  000000          SAVR1:  0                       ;R1
000554  000000          SAVR2:  0                       ;R2
000556  000000          SAVR3:  0                       ;R3
000560  000000          SAVR4:  0                       ;R4
000562  000000          SAVR5:  0                       ;R5
000564  000000          SAVSP:  0                       ;SP
000566  000000          SAVPC:  0                       ;PC
000570  000000          SAVPS:  0                       ;CONDITION CODES.
000572                  ELAST:                          ;ENTRY ABOVE IS LAST PRINTED IN PNTABL

000572  000000          MEMORY: 0                       ;MAXIMUM MEMORY FOUND BY SIZING ROUTINE
000574  000000          PASSN1: 0                       ;HOLDS RUNNING PASSNO COUNT.
000576  000000          PASSNO: 0                       ;NO. OF PASSES OF INST. TEST BEFORE PASS UPDATED.


                        .SBTTL DEVICE SELECTED CAUSED TRAP TO LOCATION 4

000600  000000          NODEVIC:        HALT            ;DEVICE SELECTED CAUSED BUS ERROR TRAP
```

```
                                ;*****************************************************************
                                .SBTTL PROGRAM INITIALIZATION AND DEVICE SETUP
                                ;REGISTER USAGE IN DEVICE SETUP
                                ;R0= SCRATCH
                                ;R1= SCRATCH
                                ;R2= HOLDS SWREG VALUE FOR DEVICE SETUP.
                                ;R3= 100 ;CONSTANT FOR INTERRUPT ENABLES
                                ;R4= 101 ;CONSTANT FOR INTERRUPT ENABLE AND GO.
                                ;*****************************************************************

000602  012706  017474          START:  MOV     #STACK,SP       ;SETUP STACK POINTER
000606  012737  000602  000532          MOV     #START,RETURN   ;IN CASE  ERROR TYPED OUT BEFORE BEGIN
000614  005037  000524                  CLR     PASS            ;INIT PASS COUNTER
000620  005037  000526                  CLR     ERROR           ;INIT ERROR COUNTER
000624  005037  000530                  CLR     CHECK           ;INIT DEVICE INTERRUPT CHECKER
000630  005037  000516                  CLR     RXSOFT          ;INIT RX FLOPPY SOFT ERROR COUNT
                                ;SETUP LOC "CHAIN" FOR RXDP CHAIN MODE OPERATION
000634  005037  000540                  CLR     CHAIN
000640  105737  000042                  TSTB    @#42            ;POSSIBLE CHAIN MODE?
000644  001406                          BEQ     1$              ;BRANCH IF NO
000646  023727  000042  015516          CMP     @#42,#$ENDAD    ;IS IT CHAIN MODE?
000654  001404                          BEQ     ESTART          ;BRANCH IF NO (MUST BE ACT11 AUTOMODE)
000656  005237  000540                  INC     @#CHAIN         ;SET "CHAIN" MODE OPERATION FLAG
000662  104001  017210          1$:     TYPE    ,MREV           ;PRINT REV

000666  005037  016212          ESTART: CLR     ICOUNT          ;INIT ITERATION COUNTER.
000672  012706  017474                  MOV     #STACK,SP       ;SET UP STACK
000676  005037  015532                  CLR     TRPB            ;INIT T BIT PASS FLAG
000702  005037  000574                  CLR     PASSN1          ;INIT RUNNING PASS COUNT
000706  005037  000576                  CLR     PASSNO          ;INIT NO. OF PASSSES BEFORE PASS COUNT PRINTED
000712  005037  016214                  CLR     SCOPEF




000716  106427  000200                  MTPS    #200            ;LOCK OUT INTERRUPTS


                                ;*****************************************************************
                                .SBTTL CHECK FOR HARDWARE SWITCH REGISTER
                                ;*****************************************************************
000722  012737  000736  000004          MOV     #1$,@#4         ;SETUP TIMEOUT VECTOR
000730  005777  014200                  TST     @SWR            ;TRY TO REFERENCE HARDWARE SR.
000734  000403                          BR      2$              ;BRANCH IF HARDWARE SR. PRESENT (I.E. NO TRAP OCC.)
000736  012737  000176  015134  1$:     MOV     #SWREG,SWR      ;POINT TO SOFTWARE SWITCH REGISTER
000744  012737  000600  000004  2$:     MOV     #NODEVIC,4      ;NON EXISTENT DEVICE TRAP
000752  012737  000340  000006          MOV     #340,6          ;DON'T ALLOW INTERRUPT
000760  012706  017474                  MOV     #STACK,SP       ;RESTORE STACK POINTER.
000764  017702  014144                  MOV     @SWR,R2         ;SAVE SWITCHES
000770  012703  000100                  MOV     #100,R3         ;INTERRUPT ENABLE
000774  012704  000101                  MOV     #101,R4         ;INTERRUPT ENABLE AND GO

001000  050377  177430          START2: BIS     R3,@TPS
001004  000005                          RESET
001006  030377  177422                  BIT     R3,@TPS         ;INTERRUPT ENABLE
```

```
1046  001012  001401                          BEQ     .+4
1047  001014  104000                          HLT                     ;*ERROR* RESET DID NOT CLEAR INTERRUPT ENABLE
1048
1049
1050                                  ;DOES "RESET" ON THE BUS LAST TOO LONG
1051  001016  012706  017474                   MOV     #STACK,SP       ;SET UP STACK
1052  001022  000005                          RESET
1053  001024  050377  177404                   BIS     R3,@TPS         ;SET INTERRUPT ENABLE
1054  001030  030377  177400                   BIT     R3,@TPS         ;IS IT SET
1055  001034  001001                          BNE     .+4             ;BRANCH IF YES
1056  001036  104000                          HLT                     ;*ERROR* RESET IS ON BUS TOO LONG
1057
1058                                  ;SETUP CONSOLE OUTPUT TEST
1059
1060  001040  005037  002130          ASKCON: CLR     TPBDAT          ;INIT CONSOLE OUTPUT DATA
1061  001044  012737  000017  002132           MOV     #15.,TPBNUL     ;INIT 15 BLANK LINES OUTPUT.
1062  001052  032702  000004                   BIT     #BIT02,R2       ;CONSOLE OUTPUT TEST WANTED?
1063  001056  001402                          BEQ     ASKASR          ;YES LEAVE INTERRUPT ON
1064  001060  040377  177350                   BIC     R3,@TPS         ;CLEAR CONSOLE OUTPUT INTERUPT ENABLE
1065
1066
1067
1068
1069
1070
1071
1072
1073
```

```
1075                                        ;SETUP ASR 33 LOW SPEED READER TEST
1076
1077  001064  012737  000001  002206  ASKASR: MOV     #1,HSRDAT       ;INIT BASE DATA FOR HSR TEST
1078  001072  005037  002300                CLR     HSPDAT          ;INIT BASE DATA FOR HSP TEST
1079  001076  005037  002304                CLR     DELAY           ;FOR READER STALL - HSR -
1080  001102  005037  002060                CLR     KBBDAT          ;INIT BASE DATA FOR SLUU  SERVICE ROUTINE
1081  001106  050377  177316                BIS     R3,@TKS         ;SET SLU INIT ENABLE
1082  001112  030227  000100                BIT     R2,#BIT06       ;SLU LOW SPEED READER INPUT TEST WANTED?
1083  001116  001004                        BNE     ASKHSP          ;BRANCH IF NO
1084  001120  005237  002060                INC     KBBDAT          ;SETUP BASE DATA FOR SLU TEST
1085  001124  050477  177300                BIS     R4,@TKS         ;AND SET READER ENABLE
1086                                        ;SETUP HSP/HSR TESTING
1087
1088  001130  005737  000546  ASKHSP: TST     HSPINH          ;HSP TEST WANTED?
1089  001134  001006                        BNE     ASKHSR          ;BRANCH IF NO
1090  001136  005777  177302                TST     @HPCSR          ;IS HSP OUT OF TAPE?
1091  001142  100001                        BPL     1$              ;BRANCH IF NO
1092  001144  104000                        HLT                     ;HSP SELECTED BUT OUT OF TAPE
1093
1094  001146  050377  177272  1$:     BIS     R3,@HPCSR       ;HSP
1095  001152  005737  000544  ASKHSR: TST     HSRINH          ;HSR TEST WANTED?
1096  001156  001010                        BNE     SETLK           ;BRANCH IF NO
1097  001160  005777  177254                TST     @HRCSR          ;IS THE TEST PATTERN IN HSR?
1098  001164  100001                        BPL     1$              ;BRANCH IF YES
1099  001166  104000                        HLT                     ;HSR SELECTED BUT TAPE NOT IN READER.
1100
1101  001170  010337  002304  1$:     MOV     R3,DELAY        ;FOR STALL HSR
1102  001174  050477  177240                BIS     R4,@HRCSR       ;HSR
1103  001200  005037  003576  SETLK:  CLR     TIME            ;INIT LINE CLOCK TIMER
1104
1105
1106                                        ;SETUP RX FLOPPY TESTING
1107
1108  001204  010200          ASKRX:  MOV     R2,R0           ;GET SR SETTINGS
1109  001206  005100                        COM     R0              ;MAKE RX INHIBITS INTO ENABLES FOR CHECK
1110  001210  042700  177774                BIC     #^C<BIT01+BIT00>,R0     ;MASK ALL BUT RX FLOPPY BITS
1111  001214  001507                        BEQ     ASKPLU          ;BRANCH IF RX FLOPPY NOT WANTED.
1112
1113  001216  005037  000516  SETRX:  CLR     RXSOFT          ;INIT RXV11 ERROR REGISTER
1114  001222  005037  000472                CLR     RXDAT           ;INIT RX DATA REGISTER
1115  001226  005037  000462                CLR     RXUN0           ;INIT RX UNIT 0 SELECT FLAG
1116  001232  005037  000464                CLR     RXUN1           ;INIT RX UNIT 1 SELECT FLAG
1117  001236  030227  000001                BIT     R2,#BIT00       ;UNIT 0 WANTED?
1118  001242  001002                        BNE     1$              ;BRANCH IF NO
1119  001244  005237  000462                INC     RXUN0           ;SET UNIT 0 SELECT FLAG
1120  001250  030227  000002  1$:     BIT     R2,#BIT01       ;UNIT 1 SELECTED?
1121  001254  001002                        BNE     2$              ;BRANCH IF NO
1122  001256  005237  000464                INC     RXUN1           ;SET UNIT 1 SELECT FLAG
1123  001262  013737  000470  000500  2$:     MOV     RXTAB,RXTA      ;SETUP TRACK BEGIN
1124  001270  013737  000466  000476          MOV     RXSAB,RXSA      ;SETUP SECTOR BEGIN
1125  001276  012737  002742  002644          MOV     #RXFRST,RXCON   ;RXFRST IS FIRST INTERRUPT SERVICE ROUTINE
1126  001304  013700  000460                MOV     RXVECP,R0       ;SET R0 POINT TO RX INTERRUPT VECTOR
1127  001310  012720  002606                MOV     #RXINT,(R0)+    ;SETUP INTERRUPT VECTOR ROUTINE
1128  001314  012710  000340                MOV     #340,(R0)       ;SETUP VECTOR PRIORITY LEVEL
1129  001320  105737  000462                TSTB    RXUN0           ;USER WANT UNIT 0?
1130  001324  001422                        BEQ     3$              ;BRANCH IF NO
```

```
1131  001326  122737  000010  000041          CMPB    #10,@#41        ;LOADED BY ACT11 OR RXDP?
1132  001334  001004                          BNE     5$              ;BRANCH IF NO
1133  001336  005037  000462                  CLR     RXUNO           ;STOP RX UNIT 0 TESTING
1134  001342  104000                          HLT                     ;*OPERATOR ERROR* UNIT 0 SELECTED FOR TEST
1135                                                                  ;BUT LOADED BY RXDP.
1136                                                                  ;SET LOCATION 41=0 AND RESTART WITH SCRATCH,
1137                                                                  ;DISKETTE IN UNIT 0.
1138  001344  000412                          BR      3$              ;AND SKIP SETTING UP UNIT 0
1139
1140
1141  001346  012777  000113  177100  5$:     MOV     #113,@RXCSR     ;READ STATUS UNIT 0
1142  001354  004737  003522                  JSR     PC,RXWTDN       ;WAIT FOR RX DONE FLAG
1143  001360  104000                          HLT                     ;*ERROR* RX UNIT 0 SELECTED BUT DONE NOT SET
1144
1145  001362  105777  177070                  TSTB    @RXDB           ;IS UNIT 0 READY?
1146  001366  100401                          BMI     .+4             ;BRANCH IF YES
1147  001370  104000                          HLT                     ;*ERROR* RX- UNIT 0 SELECTED BUT NOT READY.
1148
1149  001372  105737  000464          3$:     TSTB    RXUN1           ;USER WANT UNIT 1?
1150  001376  001412                          BEQ     4$              ;BRANCH IF NO
1151  001400  012777  000133  177046          MOV     #133,@RXCSR     ;READ STATUS UNIT 1
1152  001406  004737  003522                  JSR     PC,RXWTDN       ;WAIT FOR RX DONE FLAG
1153  001412  104000                          HLT                     ;*ERROR* RX UNIT 0 SELECTED BUT DONE NOT SET
1154
1155  001414  105777  177036                  TSTB    @RXDB           ;IS UNIT 1 READY?
1156  001420  100401                          BMI     .+4             ;BRANCH IF YES
1157  001422  104000                          HLT                     ;*ERROR* RX- UNIT 1 SELECTED BUT NOT READY
1158
1159  001424  005037  000474          4$:     CLR     RXUTT           ;SELECT UNIT 0 FOR FIRST TEST PASS.
1160  001430  005037  000520                  CLR     RXRTRY          ;INIT RX RETRY ERROR COUNT
1161                                                                  ;RX SHOULD INTERRUPT SINCE DONE SET BY SELECT UNIT COMMA
1162
1163
1164
1173
1174
1175
```

```
1177  001434  012737  000600  000004  ASKPLU: MOV     #NODEVIC,@#4    ;RESTORE NON EXIST DEVICE TRAP
1178  001442  032702  000020          BIT     #BIT04,R2       ;PLU TEST WANTED?
1179  001446  001021                  BNE     ASKEIS          ;BRANCH IF NO
1180
1181                                  ;*****************************************************************
1182                                  .SBTTL SETUP PLU TESTING **REQUIRES TEST CABLE**
1183                                  ;*****************************************************************
1184
1185  001450  013700  000426          SETPLU: MOV     @#DRVECA,R0     ;SETUP FLOATING PLU INTERRUPT VECTOR ADDRESS
1186  001454  012720  002352          MOV     #PLUA,(R0)+     ;INIT INT, VECTOR A ROUTINE
1187  001460  005020                  CLR     (R0)+           ;INIT VECTOR A PRIORITY=0
1188  001462  012720  002306          MOV     #PLUB,(R0)+     ;SETUP VECTOR B INTERRUPT .
1189  001466  005010                  CLR     (R0)            ;INIT VECTOR B PRIORITY=0
1190  001470  012700  125252          MOV     #125252,R0      ;FIRST TEST PATTERN
1191  001474  010037  000534          MOV     R0,PLUDAT       ;COMPARE LOCATION
1192  001500  010077  176716          MOV     R0,@DROBUF      ;SEND PLU TEST PATTERN
1193  001504  012777  000042  176706  MOV     #BIT05!BIT01,@DRCSR     ;ENABLE VECTOR B INTERRUPT AND REQUEST B
1194
1195                                  ;SETUP EIS/FIS TESTING
1196
1197  001512  032702  000010          ASKEIS: BIT     #BIT03,R2       ;EIS/FIS TEST WANTED?
1198  001516  001014                  BNE     ASKLP           ;BRANCH IF NO
1199
1200
1201  001520  012700  000010                  MOV     #10,R0          ;SET R0=ILLEGAL INST.VECTOR
1202  001524  012720  001546                  MOV     #1$,(R0)+       ;SET MUL TRAP RETURN
1203  001530  012710  000340                  MOV     #340,(R0)       ;SET MUL TRAP PRIORITY
1204  001534  070101                  MUL     R1,R1           ;DO A DUMMY MUL AND,
1205                                                          ;FALL THRU IF EIS/FIS MICROM PRESENT
1206  001536  005010                  CLR     (R0)            ;RESTORE HALT INSTRUCTION
1207  001540  012740  000012          MOV     #12,-(R0)       ;RESTORE ILLEGAL INST. TRAP VECTOR.
1208  001544  000401                  BR      ASKLP           ;AND CONTINUE CHECKING.
1209  001546  104000          1$:     HLT                     ;*ERROR* USER SELECTED EIS, BUT MICROM NOT PRESENT
1210
1211
1212                                  ;SETUP LP TESTING
1213
1214  001550  032702  000040          ASKLP:  BIT     #BIT05,R2       ;LP TEST WANTED?
1215  001554  001022                  BNE     SETREL          ;BRANCH IF NO
1216
1217  001556  012737  000137  002602  SETLP:  MOV     #137,SOLPAT     ;INIT DATA PATTERN
1218  001564  013737  000542  002604          MOV     LP80,CLINCT     ;INIT COLUMN COUNT
1219  001572  012737  000040  002600          MOV     #40,CURPAT      ;"SPACE" IS FIRST CHARACTER
1220  001600  105777  176644                  TSTB    @LPCSR          ;IS LP READY?
1221  001604  100401                  BMI     1$              ;BRANCH IF YES
1222  001606  104000                  HLT                     ;LP SELECTED BUT NOT READY.
1223  001610  012777  000014  176634  1$:     MOV     #14,@LPDBR      ;AFTER A FORMFEED IS SENT
1224  001616  050377  176626                  BIS     R3,@LPCSR       ;LP INTERRUPT ENABLE
```

```
1226                                    ;*****************************************************************
1227                                    .SBTTL CALL "RELOC" TO SETUP RELOCATION AND RUN MEMORY ADDRESS TEST.
1228                                    ;*****************************************************************
1229
1230  001622  004737  016332      SETREL: JSR     PC,RELOC        ;CHECK FOR TEST RELOCATION
1231
1232                                    ;PRINT MAXIMUM MEMORY AVAILABLE
1233
1234  001626  104001  017124              TYPE    ,MAXM           ;TYPE "MEMORY="
1235  001632  013703  000572              MOV     MEMORY,R3       ;GET HIGH MEMORY FOUND BY AUTOSIZING
1236  001636  004737  016012              JSR     PC,TYPOCT       ;TYPE IT.
1237  001642  104001  017146              TYPE    ,CR             ;TYPE CR/LF
1238
1239                                    ;SETUP PASSN1 FOR NO. OF INSTRUCTION PASSES REQUIRED
1240                                    ;BEFORE LOCATION PASS IS INCREMENTED.
1241  001646  005037  000576              CLR     PASSNO          ;INIT PASS COUNT HOLDER
1242  001652  032777  010000  013254      BIT     #BIT12,@SWR     ;T BIT TRAP TEST SELECTED?
1243  001660  001404                      BEQ     4$              ;BRANCH IF YES
1244  001662  032777  001000  013244  2$: BIT     #BIT09,@SWR     ;RELOCATION ENABLED?
1245  001670  001003                      BNE     3$              ;BRANCH IF NO
1246  001672  012737  000002  000576  4$: MOV     #2,PASSNO       ;REQUIRE EXTRA PASSES FOR TBIT/RELOCATE.
1247
1248
1249  001700  012737  000006  000004  3$: MOV     #6,4            ;FOR USER I/O PROGRAM
1250  001706  005037  000006              CLR     6               ;PUT TRAP HALT IN 6
1251
1253  001712  106427  000000              MTPS    #0              ;ALLOW INTERRUPTS
1255
1259
1260  001716  000401                      BR      .+4
1261  001720  000001              CPUTST: WAIT                    ;WAIT HERE FOR INTERRUPTS
1262  001722  037727  013206  002000      BIT     @SWR,#BIT10     ;INHIBIT PROCESSOR TEST
1263  001730  001373                      BNE     CPUTST
1264  001732  000137  003600              JMP     BEGIN
```

```
                                ;*****************************************************************
                                .SBTTL
                                .SBTTL  ***BEGIN - DEVICE INTERRUPT SERVICE ROUTINES***
                                .SBTTL  REGISTER R2 IS RESERVED FOR RX FLOPPY SERVICE ROUTINES
                                .SBTTL  SLU LOW SPEED READER INTERRUPT SERVICE ROUTINE
                                ;SLU RECEIVER VALUES 0 TO 377
                                ;*****************************************************************

001736  105777  176466    ASRINR: TSTB    @TKS            ;IS DONE SET
001742  100401                    BMI     .+4
001744  104000                    HLT                     ;*ERROR* FALSE INTERRUPT OCCURRED.

001746  005737  002060            TST     KBBDAT          ;POSSIBLE CONTROL-C?
001752  001021                    BNE     1$              ;BRANCH IF NO
001754  127727  176452  000003    CMPB    @TKB,#3         ;CONTROL-C ?
001762  001404                    BEQ     5$              ;YES GOTO CTCHLT
001764  127727  176442  000203    CMPB    @TKB,#203       ;CONTROL-C WITH PARITY?
001772  001031                    BNE     4$              ;BRANCH IF NO (RTI)
001774  011603            5$:     MOV     (SP),R3         ;GET PC AT INTERRUPT
001776  104001  017146            TYPE    ,CR             ;SEND CRLF
002002  004737  016012            JSR     PC,TYPOCT       ;TYPE PC AT INTERRUPT
002006  104001  017146            TYPE    ,CR             ;CR/LF
002012  000137  000170            JMP     @#CTCHLT        ; AND HALT.
002016  105777  176410    1$:     TSTB    @TKB            ;TEST DATA FOR LEADER
002022  001413                    BEQ     3$              ;IF LEADER GO BACK
002024  127737  176402  002060    CMPB    @TKB,KBBDAT     ;NOT LEADER TEST FOR DATA
002032  001401                    BEQ     2$
002034  104000                    HLT                     ;*ERROR* DATA ERROR OR ↑C TYPED DURING LSR TEST.

002036  105237  002060    2$:     INCB    KBBDAT          ;INCREMENT DATA
002042  001003                    BNE     3$
002044  012737  000001  002060    MOV     #1,KBBDAT       ;BASE DATA
002052  005277  176352    3$:     INC     @TKS            ;START READER
002056  000002            4$:     RTI                     ;RETURN TO MAINLINE

002060  000000            KBBDAT: XX                      ;EXPECTED DATA

                                ;*****************************************************************
                                .SBTTL  CONSOLE OUTPUT INTERRUPT SERVICE ROUTINE
                                ;SLU TRANSMITTER PRINT VALUES 0 TO 377
                                ;*****************************************************************

002062  105777  176346    TYOUTR: TSTB    @TPS            ;TEST FOR DONE
002066  100401                    BMI     .+4             ;BRANCH IF FLAG FOUND
002070  104000                    HLT                     ;*ERROR* FALSE INTERRUPT OCCURRED

002072  105737  002130            TSTB    TPBDAT          ;IN BETWEEN 0-377 SEQUENCES?
002076  001006                    BNE     1$              ;BRANCH IF NO
002100  105337  002132            DECB    TPBNUL          ;DONE 15 BLANKS?
002104  100005                    BPL     TYOUT1          ;BRANCH IF NO
002106  012737  000017  002132    MOV     #15.,TPBNUL     ;INIT 15. BLANK LINE COUNT FOR NEXT TIME
002114  105237  002130    1$:     INCB    TPBDAT          ;INCREMENT DATA
002120  013777  002130  176310  TYOUT1: MOV     TPBDAT,@TPB     ;OUTPUT TO DEVICE
002126  000002                    RTI                     ;RETURN TO MAINLINE
```

```
1322  002130  000000                  TPBDAT: XX                      ;TRANSMITTED DATA
1323  002132  000017                  TPBNUL: 15.                     ;NO. OF PRETEST BLANK LINES SENT
1324
1325                                  ;*****************************************************************
1326                                  .SBTTL  HIGH SPEED READER INTERRUPT SERVICE ROUTINE
1327                                  ;HSR SECTION VALUES 0 TO 377
1328                                  ;*****************************************************************
1329
1330  002134  105777  176300          HSRINR: TSTB    @HRCSR          ;IS DONE SET
1331  002140  100401                          BMI     .+4
1332  002142  104000                          HLT                     ;*ERROR* FALSE INTERRUPT OCCURRED
1333  002144  105777  176272                  TSTB    @HRDBR          ;TEST DATA FOR LEADER
1334
1335  002150  001413                          BEQ     HSRIN2          ;IF LEADER GO BACK
1336  002152  127737  176264  002206          CMPB    @HRDBR,HSRDAT   ;NOT LEADER TEST FOR DATA
1337  002160  001401                          BEQ     .+4
1338  002162  104000                          HLT                     ;*ERROR* DATA COMPARISON ERROR
1339  002164  105237  002206                  INCB    HSRDAT          ;INCREMENT DATA
1340  002170  001003                          BNE     HSRIN2
1341  002172  012737  000001  002206  HSRIN1: MOV     #1,HSRDAT       ;BASE DATA
1342  002200  005277  176234          HSRIN2: INC     @HRCSR          ;START READER
1343  002204  000002                          RTI                     ;RETURN TO MAINLINE
1344
1345  002206  000000                  HSRDAT: XX                      ;EXPECTED DATA
1346
1347                                  ;*****************************************************************
1348                                  .SBTTL  HIGH SPEED PUNCH INTERRUPT SERVICE ROUTINE
1349                                  ;HS PUNCH SECTION, VALUES 0 TO 377
1350                                  ;*****************************************************************
1351
1352                                  ;ENABLE READER ON FIX COUNT OF PUNCH ONLY (14 TIMES)
1353  002210  005037  002300          HPOUT:  CLR     HSPDAT          ;INITAL DATA
1354  002214  013777  002300  176224  HPOUT1: MOV     HSPDAT,@HPDBR   ;OUTPUT TO DEVICE
1355  002222  000002                          RTI                     ;RETURN TO MAINLINE
1356  002224  105777  176214          HPOUTR: TSTB    @HPCSR          ;TEST FOR DONE
1357  002230  100401                          BMI     .+4             ;BRANCH IF FLAG FOUND
1358  002232  104000                          HLT                     ;*ERROR* FALSE INTERRUPT RETURN
1359  002234  043777  002304  176176          BIC     DELAY,@HRCSR    ;CLEAR HSR INTERRUPT ENABLE
1360  002242  005237  002302                  INC     INTCNT          ;COUNT INTERRUPTS
1361  002246  023727  002302  000014          CMP     INTCNT,#14      ;SAVE TO TURN READER ON?
1362  002254  001005                          BNE     HPOUT2          ;NO-NEED MORE TIME
1363  002256  005037  002302                  CLR     INTCNT          ;YES RESET COUNTER
1364  002262  053777  002304  176150          BIS     DELAY,@HRCSR    ;SET READER INT ENABLE
1365  002270  105237  002300          HPOUT2: INCB    HSPDAT          ;INCREMENT DATA
1366  002274  001745                          BEQ     HPOUT           ;AT UPPER LIMIT START OVER
1367  002276  000746                          BR      HPOUT1          ;FINISH REST OF DATA
1368
1369  002300  000000                  HSPDAT: XX
1370  002302  000000                  INTCNT: 0
1371  002304  000000                  DELAY:  0                       ;EQUAL 100 IF HSR RUNNING
1372
1373                                  ;*****************************************************************
1374                                  .SBTTL  PLU INTERRUPT SERVICE ROUTINES
1375                                  ;PLUB IS THE B VECTOR SERVICE ROUTINE AND IS CALLED ONLY ONCE.
1376                                  ;*****************************************************************
1377
```

```
1378  002306  005777  176106          PLUB:   TST     @DRCSR          ;REQUEST B ASSERTED?
1379  002312  100401                  BMI     .+4             ;BRANCH IF YES
1380  002314  104000                  HLT                     ;*ERROR* PLU INTERRUPT OCCURRED
1381                                                          ;WITH REQUEST B OFF.
1382
1383  002316  023777  000534  176100          CMP     PLUDAT,@DRIBUF  ;DATA TRANSMIT EQUAL
1384                                                          ;DATA RECEIVED?
1385  002324  001401                  BEQ     .+4             ;BRANCH IF YES
1386  002326  104000                  HLT                     ;*ERROR* PLU OUTPUT BUFFER
1387                                                          ;DIDN'T TRANSMIT TO INPUT BUFFER.
1388
1389  002330  005137  000534          COM     PLUDAT          ;SET NEW TEST PATTERN =252525
1390  002334  013777  000534  176060  MOV     PLUDAT,@DROBUF  ;AND SEND IT TO PLU
1391  002342  012777  000101  176050  MOV     #BIT06!BIT00,@DRCSR     ;CLR INTERRUPT ENB B.
1392                                                          ;SET INTERRUPT ENB A AND REQUEST A.
1393
1394  002350  000002                  RTI
1395
1396                          ;*********************************************************************
1397                          ;PLUA IS THE A VECTOR SERVICE ROUTINE AND IS ENABLED AT EACH
1398                          ;"SCOPE" CALL.
1399                          ;*********************************************************************
1400
1401  002352  105777  176042  PLUA:   TSTB    @DRCSR          ;REQUEST A ASSERTED?
1402  002356  100401                  BMI     .+4             ;BRANCH IF YES
1403  002360  104000                  HLT                     ;*ERROR* PLU INTERRUPT OCCURRED
1404                                                          ;WITH REQUEST A OFF.
1405  002362  042777  000010  176030          BIC     #BIT03,@DRCSR   ;INTERRUPT A ENABLE WILL BE
1406                                                          ;SET IN SCOPEG. ROUTINE
1407  002370  023777  000534  176026          CMP     PLUDAT,@DRIBUF  ;DATA SENT TO PLU
1408                                                          ;EQUAL TO RECEIVED?
1409  002376  001401                  BEQ     .+4             ;BRANCH IF YES
1410  002400  104000                  HLT                     ;*ERROR* PLU DATA FAILED
1411  002402  005237  000534          INC     PLUDAT          ;MAKE NEW TEST PATTERN
1412  002406  052737  000020  000530  BIS     #BIT04,CHECK    ;TELL CHECK ROUTINE  PLU ALT 1'S AND 0'S PASSED
1413  002414  013777  000534  176000  MOV     PLUDAT,@DROBUF  ;AND SEND TO PLU
1414  002422  000002                  RTI
1415
1416
1417
1418                          ;*********************************************************************
1419                          .SBTTL  LINE PRINTER INTERRUPT SERVICE ROUTINE
1420                          ;LP INTERRUPT VECTOR IS 200
1421                          ;*********************************************************************
1422
1423
1424  002424  105777  176020  LPINTR: TSTB    @LPCSR          ;TEST FOR FLAG
1425  002430  100401                  BMI     .+4
1426  002432  104000                  HLT                     ;*ERROR* FALSE INTERRUPT OCCURRED
1427                                                          ;LP INTERRUPTED BUT READY WAS NOT SET
1428
1429  002434  023737  002604  000542          CMP     CLINCT,@#LP80   ;TEST FOR END OF LINE,
1430  002442  002015                  BGE     2$              ;GO GENERATE CR/LF
1431  002444  005237  002604          INC     CLINCT          ;INCREMENT LINE POSITION COUNT
1432  002450  023727  002600  000137  CMP     CURPAT,#137     ;TEST FOR MAXIMUM PATTERN
1433  002456  001403                  BEQ     1$              ;YES - GO TO LP3 AND RESET
1434  002460  005237  002600          INC     CURPAT          ;NO - INCREMENT TO NEXT PATTERN
```

```
1435  002464  000433                          BR      4$              ;GO SEND IT TO LINE PRINTER
1436  002466  012737  000040  002600  1$:     MOV     #40,CURPAT      ;RESET PATTERN AND SEND TO PRINTER
1437  002474  000427                          BR      4$              ;SENT TO LINE PRINTER
1438  002476  023737  002604  000542  2$:     CMP     CLINCT,@#LP80   ;TIME FOR THE CR?
1439  002504  003006                          BGT     3$              ;BRANCH IF YES
1440  002506  005237  002604                  INC     CLINCT          ;SET CR FLAG VIA COLUMN COUNT
1441  002512  012777  000012  175732          MOV     #12,@LPDBR      ;SEND LINEFEED TO LP
1442  002520  000426                          BR      LPEX            ;RETURN TO MAINLINE TEST
1443  002522  005037  002604          3$:     CLR     CLINCT          ;RESET COLUMN COUNT
1444  002526  012777  000015  175716          MOV     #15,@LPDBR      ;SEND CR TO LP
1445  002534  023727  002602  000137          CMP     SOLPAT,#137     ;END OF PATTERN?
1446  002542  002410                          BLT     5$              ;BRANCH IF NO
1447  002544  012737  000040  002602          MOV     #40,SOLPAT      ;INIT START OF LINE
1448  002552  000406                          BR      6$              ;AND EXIT
1449
1450  002554  013777  002600  175670  4$:     MOV     CURPAT,@LPDBR   ;SEND A CHARACTER TO LP.
1451  002562  000405                          BR      LPEX            ;RTI
1452  002564  005237  002602          5$:     INC     SOLPAT          ;INC. START OF CHARS
1453  002570  013737  002602  002600  6$:     MOV     SOLPAT,CURPAT   ;RESET CURRENT PATTERN
1454  002576  000002                  LPEX:   RTI                     ;RETURN TO MAINLINE TEST
1455
1456  002600  000000                  CURPAT: 0                       ;CURRENT CHARACTER BEING PRINTED
1457  002602  000000                  SOLPAT: 0                       ;START OF LINE CHARACTER
1458  002604  000000                  CLINCT: 0                       ;POSITION OF LINE
1460
1461
1462
1463
1464                                  ;*****************************************************************
1465                                  .SBTTL  RX FLOPPY INTERRUPT SERVICE ROUTINES
1466
1467                                  ;       ALL RX INTERRUPTS ENTER AT RXINT FOR FALSE INTERRUPT CHECKING AND THEN
1468                                  ;DISPATCH TO THE CURRENT SERVICE ROUTINE AT RXCON.
1469                                  ;       RXCON WILL CONTAIN THE FOLLOWING SERVICE ROUTINES IN SEQUENCE:
1470                                  ;RXFRST; FILL WRITE BUFFER.
1471                                  ;RXWSEC; WRITE BUFFER TO RX DISK
1472                                  ;RXRSEC; READ SECTOR
1473                                  ;RXRD;  EMPTY BUFFER AND COMPARE DATA
1474                                  ;
1475                                  ;*****************************************************************
1476
1477  002606  017702  175642          RXINT:  MOV     @RXCSR,R2       ;GET RX CSR INFO.
1478  002612  010237  000510                  MOV     R2,SRXCSR       ;SAVE CURRENT RXCS REGISTER
1479  002616  017737  175634  000506        MOV     @RXDB,SRXES     ;SAVE CURRENT ERROR STATUS
1480  002624  005702                          TST     R2              ;CHECK RX ERROR STATUS BIT
1481  002626  100407                          BMI     RXINERR         ;BRANCH IF ERROR BIT SET.
1482  002630  032702  000040                  BIT     #BIT05,R2       ;DONE BIT SET?
1483  002634  001001                          BNE     .+4             ;BRANCH IF YES
1484  002636  104000                          HLT                     ;*ERROR* RX- INTERRUPT WITH DONE NOT SET
1485
1486
1487  002640  000177  000000          1$:     JMP     @RXCON          ;GOTO SERVICE ROUTINE
1488  002644  000000                  RXCON:  XXXXX                   ;CONTAINS ADDRESS OF SERVICE ROUTINE
1489
1490                                  ;RX RESTART RX TESTING UNLESS 10 SOFT ERRORS THEN HLT.
1491
```

```
1492  002646  123727  000520  000012  RXINERR:        CMPB    RXRTRY,#10.     ;HAVE 10 SOFT ERRORS OCCURRED?
1493  002654  002401                                  BLT     1$              ;BRANCH IF NO- GIVE ANOTHER CHANCE
1494  002656  104000                                  HLT                     ;*ERROR* RX TESTING FAILED AFTER 10 RETRIES.
1495                                                                          ;SEE FOLLOW LOCATIONS FOR LAST ERROR INFO.
1496                                                                          ;SRXCSR- RX STATUS REGISTER AT LAST ERROR
1497                                                                          ;SRXES- ERROR STATUS AT LAST ERROR
1498                                                                          ;NOTE; SRXES DOES NOT INCLUDE DRIVE READY STATUS
1499
1500  002660  005237  000516          1$:     INC     RXSOFT          ;UPDATE ERROR COUNT
1501  002664  105237  000520                  INCB    RXRTRY          ;UPDATE RETRY COUNTER
1502  002670  012737  002742  002644          MOV     #RXFRST,RXCON   ;START AT FIRST TEST ON INTERRUPT.
1503  002676  052777  040000  175550          BIS     #BIT14,@RXCSR   ;RX INITIALIZE.
1504  002704  004737  003522                  JSR     PC,RXWTDN       ;RX- WAIT FOR DONE FLAG
1505  002710  104000                          HLT                     ;*ERROR* RX- DONE FLAG FAILED TO SET
1506
1507  002712  052777  000100  175534          BIS     #BIT06,@RXCSR   ;SET RX INTERRUPT ENABLE
1508  002720  005037  000472                  CLR     RXDAT           ;INIT RX DATA REGISTER
1509  002724  013737  000470  000500          MOV     RXTAB,RXTA      ;RESET BEGINNING TRACK ADDRESS
1510  002732  013737  000466  000476          MOV     RXSAB,RXSA      ;RESET BEGINNING SECTOR ADDRESS
1511  002740  000002                          RTI                     ;RETRURN TO RXFRST ON NEXT RX INTERRUPT
1512
1513
1514
1515
1516
1517                                  ;*************************************************************************
1518                                  ;RXFRST- RX FLOPPY INTERRUPT SERVICE ROUTINE
1519                                  ;       CALLED FIRST TIME AND AFTER EACH DRIVE TESTED.
1520                                  ;*************************************************************************
1521
1522
1523  002742  005737  000474          RXFRST: TST     RXUTT           ;UNIT 0 TEST TIME?
1524  002746  001405                          BEQ     1$              ;BRANCH IF YES
1525  002750  005737  000464                  TST     RXUN1           ;UNIT 1 ON LINE?
1526  002754  001005                          BNE     2$              ;BRANCH IF YES
1527  002756  005037  000474                  CLR     RXUTT           ;NO-SELECT UNIT 0 BY DEFAULT
1528  002762  005737  000462          1$:     TST     RXUN0           ;UNIT 0 ON LINE?
1529  002766  001003                          BNE     3$              ;BRANCH  IF YES
1530  002770  012737  000020  000474  2$:     MOV     #BIT04,RXUTT    ;SELECT UNIT 1 UNDER TEST.
1531
1532                                  ;FILL RX WRITE BUFFER
1533
1534  002776  012702  000200          3$:     MOV     #128.,R2        ;SIZE OF RX BUFFER
1535  003002  012777  000101  175444          MOV     #101,@RXCSR     ;SEND RX FILL BUFFER COMMAND+ INT. ENABLE
1536  003010  004737  003500          4$:     JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY FLAG
1537  003014  104000                          HLT                     ;*ERROR* RX- TRANSFER READY FLAG FAILED TO SET
1538
1539  003016  113777  000472  175432          MOVB    RXDAT,@RXDB     ;LOAD RX BUFFER WITH DATA PATTERN
1540  003024  105237  000472                  INCB    RXDAT           ;UPDATE WRITE DATA PATTERN
1541  003030  077211                          SOB     R2,4$           ;LOOP UNTIL RX BUFFER FULL
1542  003032  012737  003042  002644          MOV     #RXWSEC,RXCON   ;SET NEXT INTERRUPT ROUTINE= RXWSEC
1543  003040  000002                          RTI                     ;RETURN TO RXWSEC ON NEXT RX INTERRUPT
1544
1545                                  ;*************************************************************************
1546                                  ;RXWSEC- RX INTERRUPT SERVICE ROUTINE
1547                                  ;       CALLED AFTER RXFRST AND ITSELF UNTIL EVERY 3RD SECTOR ON TRACKS 1
```

```
                                        ;AND 114 ARE WRITTEN WITH INCREMENTING DATA PATTERN.
                                        ;*****************************************************************

003042  013702  000474          RXWSEC: MOV     RXUTT,R2        ;SELECT DRIVE
003046  052702  000105                  BIS     #105,R2         ;SELECT WRITE SECTOR +INTERRUPT ENABLE
003052  010277  175376                  MOV     R2,@RXCSR       ;LOAD INTO RX COMMAND REGISTER.
003056  004737  003500                  JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY
003062  104000                          HLT                     ;*ERROR* RX- TRANSFER READY FAILED TO SET

003064  013777  000476  175364          MOV     RXSA,@RXDB      ;LOAD RX SECTOR ADDRESS
003072  004737  003500                  JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY
003076  104000                          HLT                     ;*ERROR* RX- TRANSFER READY FAILED TO SET

003100  013777  000500  175350          MOV     RXTA,@RXDB      ;LOAD RX TRACK ADDRESS AND INTERRUPT WHEN DONE.

                                                                ;UPDATE RX SECTOR AND TRACK ADDRESSES.

003106  012737  002742  002644          MOV     #RXFRST,RXCON   ;RETURN TO RXFRST ON NEXT INTERRUPT.
003114  062737  000003  000476          ADD     #3,RXSA         ;UPDATE SECTOR
003122  023727  000476  000031          CMP     RXSA,#31        ;IS THIS LAST SECTOR?
003130  002425                          BLT     4$              ;BRANCH IF NO (RTI)
003132  003005                          BGT     1$              ;BRANCH IF TIME TO SWITCH TRACKS
003134  023737  000470  000500          CMP     RXTAB,RXTA      ;IS THIS FIRST TRACK?
003142  001005                          BNE     2$              ;BRANCH IF NO (ALL WRITES ARE DONE)
003144  000417                          BR      4$              ;RTI

003146  012737  000114  000500  1$:     MOV     #114,RXTA       ;SET LAST TRACK ADDRESS
003154  000410                          BR      3$              ;(SET FIRST SECTOR + RTI)
003156  013737  000470  000500  2$:     MOV     RXTAB,RXTA      ;SET TRACK BEGIN ADDRESS
003164  005037  000472                  CLR     RXDAT           ;RESET INCREMENTING RX DATA PATTERN
003170  012737  003206  002644          MOV     #RXRSEC,RXCON   ;SET NEXT RX INTERRUPT ROUTINE=RXRSEC
003176  013737  000466  000476  3$:     MOV     RXSAB,RXSA      ;SET SECTOR BEGIN ADDRESS
003204  000002                  4$:     RTI                     ;RETURN TO RXWSEC OR RXRSEC ON NEXT RX INTERRUPT


                                        ;*****************************************************************
                                        ;RXRSEC- RX INTERRUPT SERVICE ROUTINE
                                        ;       CALLED AFTER RXWSEC AND RXRD UNTIL EVERY 3RD SECTOR ON
                                        ;TRACKS 1 AND 114 ARE READ INTO BUFFER.
                                        ;*****************************************************************

003206  013702  000474          RXRSEC: MOV     RXUTT,R2        ;SELECT DRIVE
003212  052702  000107                  BIS     #107,R2         ;SELECT READ COMMAND AND INTERRUPT ENABLE
003216  010277  175232                  MOV     R2,@RXCSR       ;LOAD RX COMMAND REGISTER
003222  004737  003500                  JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY
003226  104000                          HLT                     ;*ERROR* RX- TRANSFER READY FAILED TO SET

003230  013777  000476  175220          MOV     RXSA,@RXDB      ;LOAD SECTOR ADDRESS
003236  004737  003500                  JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY
003242  104000                          HLT                     ;*ERROR* RX-TRANSFER READY FAILED TO SET

003244  013777  000500  175204          MOV     RXTA,@RXDB      ;LOAD TRACK ADDRESS AND INITIATE AN INTERRUPT.
003252  012737  003262  002644          MOV     #RXRD,RXCON     ;SET NEXT RX SERVICE ROUTINE= RXRD
003260  000002                          RTI
```

```
                                ;*********************************************************************
                                ;RXRD- RX INTERRUPT SERVICE ROUTINE
                                ;       CALLED AFTER RXRSEC UNTIL ALL SECTORS ON TRACK 1 AND 114 ARE COMPARED
                                ;FOR CORRECT DATA.
                                ;       DATA IS COMPARED AS IT COMES FROM THE RX BUFFER AND IS NOT STORED
                                ;UNLESS IT IS IN ERROR.
                                ;*********************************************************************

003262  010146          RXRD:   MOV     R1,-(SP)        ;SAVE R1 ON STACK
003264  012702  000200          MOV     #128.,R2        ;SIZE OF RX BUFFER.
003270  012777  000103  175156  MOV     #103,@RXCSR     ;LOAD RX READ BUFFER CMD. + INT. ENABLE

                        ;*BEGIN READ RX BUFFER LOOP*
003276  004737  003500  1$:     JSR     PC,RXWTTR       ;RX- WAIT TRANSFER READY
003302  104000                  HLT                     ;*ERROR* RX- TRANSFER READY FAILED TO SET

003304  117701  175146          MOVB    @RXDB,R1        ;GET RX BUFFER DATA
003310  120137  000472          CMPB    R1,RXDAT        ;IS DATA OK?
003314  001401                  BEQ     .+4             ;BRANCH IF YES
003316  104000                  HLT                     ;*ERROR* RX DATA BUFFER COMPARE ERROR.

003320  105237  000472          INCB    RXDAT           ;UPDATE COMPARE DATA
003324  077214                  SOB     R2,1$           ;LOOP UNTIL BUFFER EMPTY
                        ;*END READ RX BUFFER LOOP*
003326  012601                  MOV     (SP)+,R1        ;RESTORE R1

003330  005037  000520          CLR     RXRTRY          ;INIT RETRY COUNTER
                        ;UPDATE SECTOR AND TRACK ADDRESSES

003334  012737  003206  002644          MOV     #RXRSEC,RXCON   ;GOTO RXRSEC ON NEXT INTERRUPT FROM RX
003342  062737  000003  000476          ADD     #3,RXSA         ;UPDATE SECTOR ADDRESS
003350  023727  000476  000031          CMP     RXSA,#31        ;IS NEXT SECTOR LAST?
003356  002446                  BLT     4$              ;BRANCH IF NO (RTI)
003360  003037                  BGT     3$              ;BRANCH IF TIME TO SWITCH TRACKS
003362  023737  000470  000500          CMP     RXTAB,RXTA      ;IS THIS FIRST TRACK?
003370  001441                  BEQ     4$              ;BRANCH IF YES (RTI)
                        ;SELECT DRIVE FOR NEXT TEST PASS
003372  013737  000466  000476          MOV     RXSAB,RXSA      ;RESET TO BEGIN SECTOR
003400  013737  000470  000500          MOV     RXTAB,RXTA      ;RESET TO BEGIN TRACK
003406  012737  002742  002644          MOV     #RXFRST,RXCON   ;SET NEXT RX SERVICE ROUTINE=RXFRST
003414  005037  000472                  CLR     RXDAT           ;RESET RX DATA REGISTER
003420  005737  000474                  TST     RXUTT           ;JUST TESTED UNIT 0?
003424  001406                  BEQ     2$              ;BRANCH IF YES
003426  052737  000002  000530          BIS     #BIT01,CHECK    ;TELL WORLD RX UNIT 1 MADE A PASS
003434  005037  000474                  CLR     RXUTT           ;TEST UNIT 0 NEXT PASS
003440  000415                  BR      4$              ;RTI
003442  052737  000040  000474  2$:     BIS     #BIT05,RXUTT    ;TEST UNIT 1 NEXT PASS
003450  052737  000001  000530          BIS     #BIT00,CHECK    ;TELL WORLD RX UNIT 0 MADE A PASS
003456  000406                  BR      4$              ;RTI
003460  013737  000466  000476  3$:     MOV     RXSAB,RXSA      ;RESET SECTOR FOR LAST TRACK TEST
003466  012737  000114  000500          MOV     #114,RXTA       ;SET LAST TRACK ADDRESS
003474  000002          4$:     RTI                     ;RETURN TO RXRSEC OR RXFRST ON NEXT RX INTERRUPT
```

```
                                ;*********************************************************************
                                ;RXWTTR- RX SUBROUTINE TO WAIT FOR TRANSFER READY.
                                ;RXWTTR TIMESOUT AFTER ABOUT 1 SEC.
                                ;CALL IS;
                                ;       JSR     PC,RXWTTR
003476  104000                  ;       HLT                             ;*ERROR* ERROR RETURN
                                ;*********************************************************************

003500  005037  000502          RXWTTR: CLR     RXLCNT                  ;INIT WATCHDOG COUNTER
003504  105777  174744          1$:     TSTB    @RXCSR                  ;RX- TRANSFER READY SET?
003510  100424                          BMI     RXWRET                  ;BRANCH IF YES (EXIT GOOD RETURN)
003512  005237  000502                  INC     RXLCNT                  ;INCR. WATCHDOG COUNTER
003516  001372                          BNE     1$                      ;BRANCH IF NOT OVERFLOWED
003520  000422                          BR      RXWERR                  ;WATCHDOG TIMED OUT TAKE ERROR RETURN


                                ;*********************************************************************
                                ;RXWTDN- RX SUBROUTINE TO WAIT FOR DONE FLAG
                                ;       RXWTDN WAITS FOR UP TO A MINUTE FOR THE DONE FLAG TO SET
                                ;CALL IS;
                                ;       JSR     PC,RXWTDN
                                ;       HLT                             ;*ERROR* ERROR RETURN
                                ;*********************************************************************

003522  005037  000502          RXWTDN: CLR     RXLCNT                  ;INIT LOW WATCHDOG TIMER
003526  012737  177740  000504          MOV     #177740,RXHCNT          ;INIT HIGH WATCHDOG TIMER
003534  032777  000040  174712  1$:     BIT     #BIT05,@RXCSR           ;DONE FLAG SET?
003542  001007                          BNE     RXWRET                  ;BRANCH IF YES
003544  062737  000001  000502          ADD     #1,RXLCNT                       ;INCR. LOW WATCHDOG TIMER
003552  005537  000504                  ADC     RXHCNT                  ;ADD OVERFLOW TO HIGH WATCHDOG TIMER
003556  001366                          BNE     1$                      ;BRANCH IF NOT TIMED OUT
003560  000402                          BR      RXWERR                  ;ERROR RETURN-TIMER OVERFLOWED
003562  062716  000002          RXWRET: ADD     #2,(SP)                 ;SKIP ERROR ON RETURN
003566  000207                  RXWERR: RTS     PC                      ;RETURN FROM WATCHDOG TIMER
```

```
1722
1723                                    ;*****************************************************************
1724                                    .SBTTL  LINE CLOCK INTERRUPT SERVICE ROUTINE
1725                                    ;LOCATION TIME IS PRINTED IN OCTAL AT END OF PASS
1726                                    ;*****************************************************************
1727  003570  005237  003576      CLOCK:  INC     TIME            ;UPDATE TIMER
1728  003574  000002                      RTI                     ;RETURN TO MAINLINE TEST
1729  003576  000000              TIME:   0                       ;HOLDS LINE CLOCK INTERRUPT COUNT.
1730
1731                                    ;*****************************************************************
1732                                    .SBTTL  ***END - DEVICE INTERRUPT SERVICE ROUTINES***
1733                                    .SBTTL
1734                                    ;*****************************************************************
1735
1736
1737
```

```
                                ;*****************************************************************
                                .SBTTL  ***BEGIN RELOCATED INSTRUCTION TEST***
                                .SBTTL BACKGROUND PROCESSOR INSTRUCTION TESTS
                                ;       REGISTER R2 CANNOT BE USED IN RELOCATED TESTS (RESERVED FOR FLOPPY)
                                .DSABL AMA
                                ;*****************************************************************

003600  010700                  BEGIN:  MOV     PC,R0           ;SET SCOPE RETURN TO CURRENT BANK
003602  162700  000002                  SUB     #2,R0           ;POINT TO BEGIN:
003606  010037  000532                  MOV     R0,@#RETURN     ;AND SAVE IF SCOPE LOOPING
003612  010037  000522                  MOV     R0,@#LOCATE     ;SAVE "BEGIN" ADDRESS EVEN IF RELOCATED

003616  012737  001000  016212          MOV     #1000,@#ICOUNT  ;ITERATION COUNT

                                ;*****************************************************************
                                ;TEST BRANCH INSTRUCTIONS
                                ;*****************************************************************

003624  000277                  BRXX:   SCC                     ;CC=1111
003626  103003                          BCC     1$              ;BRANCH IF C FAILED
003630  102002                          BVC     1$              ;BRANCH IF V FAILED
003632  001001                          BNE     1$              ;BRANCH IF Z FAILED
003634  100401                          BMI     .+4             ;BRANCH IF N OK
003636  104000                  1$:     HLT                     ;*ERROR* SCC OR BRANCH FAILED

003640  103401                          BCS     .+4             ;BRANCH IF C OK
003642  104000                          HLT                     ;*ERROR* BCS FAILED
003644  102401                          BVS     .+4             ;BRANCH IF V OK
003646  104000                          HLT                     ;*ERROR* BVS FAILED
003650  001401                          BEQ     .+4             ;BRANCH IF Z OK
003652  104000                          HLT                     ;*ERROR* BEQ FAILED
003654  100002                          BPL     2$              ;BRANCH IF N FAILED
003656  101001                          BHI     2$              ;BRANCH IF C OR Z FAILED
003660  101401                          BLOS    .+4             ;BRANCH IF C OR Z OK
003662  104000                  2$:     HLT                     ;*ERROR* BPL, BHI, BLOS FAILED

003664  000241                          CLC                     ;CC=1110
003666  103402                          BCS     3$              ;BRANCH IF C FAILED
003670  103401                          BLO     3$              ;BRANCH IF C FAILED
003672  103001                          BCC     .+4             ;BRANCH IF C OK
003674  104000                  3$:     HLT                     ;*ERROR* CLC,BCS,BLO,BCC OR CLC FAILED

003676  000242                          CLV                     ;CC=1100
003700  102403                          BVS     4$              ;BRANCH IF V FAILED
003702  003002                          BGT     4$              ;BRANCH IF BGT FUNCTION FAILED
003704  002001                          BGE     4$              ;BRANCH IF BGE FUNCTION FAILED
003706  003401                          BLE     .+4             ;BRANCH IF BLE FUNCTION OK
003710  104000                  4$:     HLT                     ;*ERROR* CLV,BVS,BGT,BGE OR BLE FAILED

003712  000244                          CLZ                     ;CC=1000
003714  100002                          BPL     5$              ;BRANCH IF N FAILED
003716  001401                          BEQ     5$              ;BRANCH IF Z FAILED
003720  002401                          BLT     .+4             ;BRANCH IF BLT FUNTION OK.
003722  104000                  5$:     HLT                     ;*ERROR* CLZ,BPL,BEQ,BGT FAILED
```

```
1796
1797  003724  000250                          CLN                     ;CC=0000
1798  003726  003001                          BGT     .+4             ;BRANCH IF BGT FUNCTION OK
1799  003730  104000                  BREND:  HLT                     ;*ERROR* BGT OR CLN FAILED
1800
1801
1802                                  ;*********************************************************************
1803                                  ;TEST COMPARE INSTRUCTION INDEXED
1804                                  ;*********************************************************************
1805
1806  003732  012700  177770                  MOV     #-10,R0         ;MINUS 10 TO REG 0
1807  003736  026027  015100  125252          CMP     A(0),#125252    ;(A INDEX BY MINUS 10) TO #125252
1808  003744  001401                          BEQ     .+4
1809  003746  104000                          HLT                     ;*ERROR* COMPARE WITH INDEX FAILED
1810  003750  104400                          SCOPE
1811
1812
1813
1814  003752  012700  177770                  MOV     #-10,R0         ;FOR INDEX
1815  003756  022760  125252  015100          CMP     #125252,A(0)    ;A INDEXED
1816  003764  001401                          BEQ     .+4
1817  003766  104000                          HLT                     ;*ERROR* COMPARE FAILED DESTINATION INDEX
1818  003770  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1819
1831  003772  012700  000010                  MOV     #10,R0          ;INDEX
1832  003776  026027  015100  052525          CMP     A(0),#052525
1833  004004  001401                          BEQ     .+4
1834  004006  104000                          HLT                     ;*ERROR* COMPARE FAILED
1835  004010  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1836
1837
1838
1839  004012  012700  000010                  MOV     #10,R0
1840  004016  022760  052525  015100          CMP     #052525,A(0)
1841  004024  001401                          BEQ     .+4
1842  004026  104000                          HLT                     ;*ERROR* COMPARE FAILED
1843  004030  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1844
1845
1846
1847  004032  012700  177770                  MOV     #-10,R0
1848  004036  026060  015100  015100          CMP     A(0),A(0)
1849  004044  001401                          BEQ     .+4
1850  004046  104000                          HLT                     ;*ERROR* COMPARE FAILED
1851  004050  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1852
1853
1854
1855  004052  012700  000010                  MOV     #+10,R0
1856  004056  026060  015100  015100          CMP     A(0),A(0)
1857  004064  001401                          BEQ     .+4
1858  004066  104000                          HLT                     ;*ERROR* COMPARE FAILED
1859  004070  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1860
1861
1862
```

```
1863  004072  012700  177770            MOV     #-10,R0
1864  004076  012701  000004            MOV     #+4,R1
1865  004102  026061  015100  015100    CMP     A(0),A(1)
1866  004110  001401                    BEQ     .+4
1867  004112  104000                    HLT                     ;*ERROR* COMPARE FAILED
1868  004114  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1869
1870
1871
1872  004116  026160  015100  015100    CMP     A(1),A(0)
1873  004124  001401                    BEQ     .+4
1874  004126  104000                    HLT                     ;*ERROR* COMPARE FAILED
1875  004130  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1876
1877
1878
1879  004132  012700  177774            MOV     #-4,R0
1880  004136  012701  000010            MOV     #+10,R1
1881  004142  026061  015100  015100    CMP     A(0),A(1)
1882  004150  001401                    BEQ     .+4
1883  004152  104000                    HLT                     ;*ERROR* CMP FAILED
1884  004154  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1885
1886
1887  004156  012700  177774            MOV     #-4,R0
1888  004162  012701  000010            MOV     #10,R1
1889  004166  026160  015100  015100    CMP     A(1),A(0)
1890  004174  001401                    BEQ     .+4
1891  004176  104000                    HLT                     ;*ERROR* COMPARE FAILED
1892  004200  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1893
1894
1895                          ;*************************************************************************
1896                          ;TEST MOVE ODD BYTE TO REGISTER
1897                          ;*************************************************************************
1898
1899
1900
1901
1902
1903  004202  116700  000007            MOVB    1$+1,R0         ;GET HIGH BYTE OF "HLT" INSTRUCTION
1904  004206  122700  000210            CMPB    #210,R0         ;DID R0 GET ODD BYTE?
1905  004212  001401                    BEQ     .+4             ;BRANCH IF YES
1906  004214  104000              1$:   HLT
1907  004216  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1908
1909
1910                          ;*************************************************************************
1911                          ;TEST MOVE INSTRUCTION FOR INDEX
1912                          ;*************************************************************************
1913
1914
1915  004220  012700  177770            MOV     #-10,R0
1916  004224  016067  015100  010670    MOV     A(0),TEMP
1917  004232  026727  010664  125252    CMP     TEMP,#125252
1918  004240  001401                    BEQ     .+4
```

```
1919  004242  104000                          HLT                     ;*ERROR* COMPARE FAILED
1920  004244  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1921
1922
1923
1924  004246  012700  000010                  MOV     #+10,R0
1925  004252  016067  015100  010642          MOV     A(0),TEMP
1926  004260  026727  010636  052525          CMP     TEMP,#052525
1927  004266  001401                          BEQ     .+4
1928  004270  104000                          HLT                     ;*ERROR* MOV FAILED
1929  004272  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1930
1931
1932
1933  004274  012700  177770                  MOV     #-10,R0
1934  004300  012760  125252  015122          MOV     #125252,TEMP(0)
1935  004306  023727  015112  125252          CMP     @#C,#125252
1936  004314  001401                          BEQ     .+4
1937  004316  104000                          HLT                     ;*ERROR* MOV FAILED
1938  004320  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1939
1940
1941
1942  004322  012700  000010                  MOV     #+10,R0
1943  004326  012760  052525  015122          MOV     #052525,TEMP(0)
1944  004334  023727  015132  052525          CMP     @#TEMP+10,#052525
1945  004342  001401                          BEQ     .+4
1946  004344  104000                          HLT                     ;*ERROR* MOV FAILED
1947  004346  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1948
1949
1950
1951                                  ;*************************************************************************
1952                                  ;TEST BIC INSTRUCTION FOR INDEXING
1953                                  ;*************************************************************************
1954
1955  004350  012767  177777  010544          MOV     #-1,TEMP
1956  004356  012700  177770                  MOV     #-10,R0
1957  004362  046067  015100  010532          BIC     A(0),TEMP
1958  004370  026727  010526  052525          CMP     TEMP,#052525
1959  004376  001401                          BEQ     .+4
1960  004400  104000                          HLT                     ;*ERROR* BIC FAILED
1961  004402  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1962
1963
1964
1965  004404  012767  177777  010510          MOV     #-1,TEMP
1966  004412  012700  000010                  MOV     #10,R0
1967  004416  046067  015100  010476          BIC     A(0),TEMP
1968  004424  026727  010472  125252          CMP     TEMP,#125252
1969  004432  001401                          BEQ     .+4
1970  004434  104000                          HLT                     ;*ERROR* BIC FAILED
1971  004436  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1972
1973
1974
```

```
1975  004440  012737  177777  015132          MOV     #-1,@#TEMP+10
1976  004446  012700  000010                  MOV     #10,R0
1977  004452  042760  125252  015122          BIC     #125252,TEMP(0)
1978  004460  023727  015132  052525          CMP     @#TEMP+10,#52525
1979  004466  001401                          BEQ     .+4
1980  004470  104000                          HLT                     ;*ERROR* BIC FAILED
1981  004472  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1982
1983
1984
1985  004474  012700  177770                  MOV     #-10,R0
1986  004500  012767  177777  010404          MOV     #-1,TEMP-10
1987  004506  042767  052525  010376          BIC     #052525,TEMP-10
1988  004514  026727  010372  125252          CMP     TEMP-10,#125252
1989  004522  001401                          BEQ     .+4
1990  004524  104000                          HLT                     ;*ERROR* BIC FAILED
1991  004526  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
1992
1993
1994                                  ;*************************************************************************
1995                                  ;TEST BIT INSTRUCTION MODE 0
1996                                  ;*************************************************************************
1997
1998  004530  012700  125252          BIT0:   MOV     #125252,R0      ;SET R0=ALT 1'S
1999  004534  032700  052525                  BIT     #52525,R0       ;125252 ANDED 52525=0
2000  004540  001401                          BEQ     .+4             ;BRANCH IF RESULT OK
2001  004542  104000                          HLT                     ;*ERROR* BIT MODE 0 FAILED
2002  004544  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2003
2004
2005                                  ;*************************************************************************
2006                                  ;TEST BITB INSTRUCTION MODE NON 0. (27)
2007                                  ;*************************************************************************
2008
2009  004546  112767  000200  010347  BITB0:  MOVB    #200,TEMP+1
2010  004554  132767  000200  010341          BITB    #200,TEMP+1     ;200 ANDED 200=200
2011  004562  001001                          BNE     .+4             ;BRANCH IF OK
2012  004564  104000                          HLT                     ;*ERROR* BITB MODE 27 FAILED
2013  004566  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2014
2015
2016                                  ;*************************************************************************
2017                                  ;TEST SUBTRACT INSTRUCTION FOR INDEXING
2018                                  ;*************************************************************************
2019
2020  004570  012767  125252  010324          MOV     #125252,TEMP
2021  004576  012700  177770                  MOV     #-10,R0
2022  004602  166067  015100  010312          SUB     A(0),TEMP
2023  004610  001401                          BEQ     .+4
2024  004612  104000                          HLT                     ;*ERROR* SUB FAILED
2025  004614  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2026
2027
2028
2029  004616  012737  125252  015122          MOV     #125252,@#TEMP
2030  004624  012700  177770                  MOV     #-10,R0
```

```
2031  004630  166760  010234  015132          SUB     B,TEMP+10(0)
2032  004636  001401                          BEQ     .+4
2033  004640  104000                          HLT                     ;*ERROR* SUB FAILED
2034  004642  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2035
2036
2037
2038  004644  012767  052525  010250          MOV     #052525,TEMP
2039  004652  012700  000010                  MOV     #10,R0
2040  004656  166067  015100  010236          SUB     A(0),TEMP
2041  004664  001401                          BEQ     .+4
2042  004666  104000                          HLT                     ;*ERROR* SUB FAILED
2043  004670  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2044
2045
2046
2047  004672  012737  052525  015122          MOV     #052525,@#TEMP
2048  004700  012700  000010                  MOV     #10,R0
2049  004704  166760  010200  015112          SUB     A+10,C(0)
2050  004712  001401                          BEQ     .+4
2051  004714  104000                          HLT                     ;*ERROR* SUB FAILED
2052  004716  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2053
2054
2055
2056                                  ;*************************************************************************
2057                                  ;TEST UNARYS INDEXED
2058                                  ;*************************************************************************
2059
2060  004720  012737  177777  015122          MOV     #-1,@#TEMP
2061  004726  012700  177770                  MOV     #-10,R0
2062  004732  005060  015132                  CLR     D(0)
2063  004736  005737  015122                  TST     @#TEMP
2064  004742  001401                          BEQ     .+4
2065  004744  104000                          HLT                     ;*ERROR* CLR FAILED
2066  004746  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2067
2068
2069
2070  004750  012737  177777  015122          MOV     #-1,@#TEMP
2071  004756  012700  000010                  MOV     #+10,R0
2072  004762  005060  015112                  CLR     C(0)
2073  004766  005737  015122                  TST     @#TEMP
2074  004772  001401                          BEQ     .+4
2075  004774  104000                          HLT                     ;*ERROR* CLR FAILED
2076  004776  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2077
2078
2079
2080  005000  012737  177777  015122          MOV     #-1,@#TEMP
2081  005006  012700  177770                  MOV     #-10,R0
2082  005012  005160  015132                  COM     D(0)
2083  005016  005737  015122                  TST     @#TEMP
2084  005022  001401                          BEQ     .+4
2085  005024  104000                          HLT                     ;*ERROR* COM FAILED
2086  005026  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
```

```
2087
2088
2089
2090  005030  012737  177777  015122          MOV     #-1,@#TEMP
2091  005036  012700  000010                  MOV     #10,R0
2092  005042  005160  015112                  COM     C(0)
2093  005046  005737  015122                  TST     @#TEMP
2094  005052  001401                          BEQ     .+4
2095  005054  104000                          HLT                     ;*ERROR* COM FAILED
2096  005056  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2097
2098
2099  005060  012737  177777  015122          MOV     #-1,@#TEMP
2100  005066  012700  177770                  MOV     #-10,R0
2101  005072  005260  015132                  INC     D(0)
2102  005076  005737  015122                  TST     @#TEMP
2103  005102  001401                          BEQ     .+4
2104  005104  104000                          HLT                     ;*ERROR* INC FAILED
2105  005106  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2106
2107
2108
2109  005110  012737  177777  015122          MOV     #-1,@#TEMP
2110  005116  012700  000010                  MOV     #+10,R0
2111  005122  005260  015112                  INC     C(0)
2112  005126  005737  015122                  TST     @#TEMP
2113  005132  001401                          BEQ     .+4
2114  005134  104000                          HLT                     ;*ERROR* INC FAILED
2115  005136  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2116
2117
2118
2119  005140  012737  000001  015122          MOV     #1,@#TEMP
2120  005146  012700  177770                  MOV     #-10,R0
2121  005152  005360  015132                  DEC     D(0)
2122  005156  005737  015122                  TST     @#TEMP
2123  005162  001401                          BEQ     .+4
2124  005164  104000                          HLT                     ;*ERROR* DEC FAILED
2125  005166  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2126
2127
2128
2129  005170  012737  000001  015122          MOV     #1,@#TEMP
2130  005176  012700  000010                  MOV     #0,R0
2131  005202  005360  015112                  DEC     C(0)
2132  005206  005737  015122                  TST     @#TEMP
2133  005212  001401                          BEQ     .+4
2134  005214  104000                          HLT                     ;*ERROR* DEC FAILED
2135  005216  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2136
2137
2138
2139  005220  012737  000001  015122          MOV     #1,@#TEMP
2140  005226  012700  177770                  MOV     #-10,R0
2141  005232  005460  015132                  NEG     D(0)
2142  005236  022737  177777  015122          CMP     #-1,@#TEMP
```

```
2143  005244  001401                          BEQ     .+4
2144  005246  104000                          HLT                     ;*ERROR* NEG FAILED
2145  005250  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2146
2147
2148
2149  005252  012737  000001  015122          MOV     #1,@#TEMP
2150  005260  012700  000010                  MOV     #+10,R0
2151  005264  005460  015112                  NEG     C(0)
2152  005270  022737  177777  015122          CMP     #-1,@#TEMP
2153  005276  001401                          BEQ     .+4
2154  005300  104000                          HLT                     ;*ERROR* NEG FAILED
2155  005302  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2156
2157
2158
2159  005304  012737  177777  015122          MOV     #-1,@#TEMP
2160  005312  012700  177770                  MOV     #-10,R0
2161  005316  000261                          SEC
2162  005320  005560  015132                  ADC     D(0)
2163  005324  005737  015122                  TST     @#TEMP
2164  005330  001401                          BEQ     .+4
2165  005332  104000                          HLT                     ;*ERROR* ADC FAILED
2166  005334  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2167
2168
2169
2170  005336  012737  177777  015122          MOV     #-1,@#TEMP
2171  005344  012700  000010                  MOV     #+10,R0
2172  005350  000261                          SEC
2173  005352  005560  015112                  ADC     C(0)
2174  005356  005737  015122                  TST     @#TEMP
2175  005362  001401                          BEQ     .+4
2176  005364  104000                          HLT                     ;*ERROR* ADC FAILED
2177  005366  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2178
2179
2180
2181  005370  012737  000001  015122          MOV     #1,@#TEMP
2182  005376  012700  177770                  MOV     #-10,R0
2183  005402  000261                          SEC
2184  005404  005660  015132                  SBC     D(0)
2185  005410  005737  015122                  TST     @#TEMP
2186  005414  001401                          BEQ     .+4
2187  005416  104000                          HLT                     ;*ERROR* SBC FAILED
2188  005420  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2189
2190
2191
2192  005422  012737  000001  015122          MOV     #1,@#TEMP
2193  005430  012700  000010                  MOV     #+10,R0
2194  005434  000261                          SEC
2195  005436  005660  015112                  SBC     C(0)
2196  005442  005737  015122                  TST     @#TEMP
2197  005446  001401                          BEQ     .+4
2198  005450  104000                          HLT                     ;*ERROR* SBC FAILED
```

```
2199  005452  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2200
2201
2202
2203                                  ;*****************************************************************
2204                                  ;TEST JMP INDIRECT
2205                                  ;*****************************************************************
2206
2207  005454  010700                          MOV     PC,R0
2208  005456  062700  000010                 ADD     #10,R0
2209  005462  000110                         JMP     @R0
2210  005464  104000                         HLT                     ;*ERROR* JMP FAILED
2211  005466  000240                         NOP
2212  005470  104400                         SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2213
2214
2215
2216  005472  010600                         MOV     SP,R0
2217  005474  010001                         MOV     R0,R1
2218  005476  010103                         MOV     R1,R3           ;R2 IS NOT TESTED HERE SINCE IT IS USED IN THE FLOPPY
2219
2220  005500  010304                         MOV     R3,R4
2221  005502  010405                         MOV     R4,R5
2222  005504  020605                         CMP     SP,R5
2223  005506  001401                         BEQ     .+4
2224  005510  104000                         HLT                     ;*ERROR* MOV REGISTER FAILED
2225  005512  104400                         SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2226
2227                                  .SBTTL  TEST INDIRECT ADDRESSING
2228                                  ;*****************************************************************
2229                                  ;TEST COMPARE INSTRUCTION
2230                                  ;*****************************************************************
2231
2232  005514  023727  015070  125252          CMP     @#8,#125252
2233  005522  001401                         BEQ     .+4
2234  005524  104000                         HLT                     ;*ERROR* CMP FAILED
2235  005526  104400                         SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2236
2237
2238
2239  005530  022737  125252  015070          CMP     #125252,@#8
2240  005536  001401                         BEQ     .+4
2241  005540  104000                         HLT                     ;*ERROR* CMP FAILED
2242  005542  104400                         SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2243
2244
2245
2246  005544  023737  015070  015070          CMP     @#8,@#8
2247  005552  001401                         BEQ     .+4
2248  005554  104000                         HLT                     ;*ERROR* CMP FAILED
2249  005556  104400                         SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2250
2251
2252
2253                                  ;*****************************************************************
2254                                  ;TEST MOV INSTRUCTIONS
```

```
2255                                    ;*************************************************************************
2256
2257  005560  013700  015070                      MOV     @#B,R0
2258  005564  022700  125252                      CMP     #125252,R0
2259  005570  001401                              BEQ     .+4
2260  005572  104000                              HLT                     ;*ERROR* MOV FAILED
2261  005574  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2262
2263
2264
2265  005576  012737  125252  015122              MOV     #125252,@#TEMP
2266  005604  023737  015070  015122              CMP     @#B,@#TEMP
2267  005612  001401                              BEQ     .+4
2268  005614  104000                              HLT                     ;*ERROR* MOV FAILED
2269  005616  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2270
2271
2272
2273  005620  013737  015070  015112              MOV     @#B,@#C
2274  005626  023737  015070  015112              CMP     @#B,@#C
2275  005634  001401                              BEQ     .+4
2276  005636  104000                              HLT                     ;*ERROR* MOV FAILED
2277  005640  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2278
2279
2280                                    ;*************************************************************************
2281                                    ;TEST BIC INSTRUCTION INDIRECT
2282                                    ;*************************************************************************
2283
2284  005642  012700  177777                      MOV     #-1,R0
2285  005646  043700  015070                      BIC     @#B,R0
2286  005652  020027  052525                      CMP     R0,#052525
2287  005656  001401                              BEQ     .+4
2288  005660  104000                              HLT                     ;*ERROR* BIC FAILED
2289  005662  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2290
2291
2292
2293  005664  012737  177777  015122              MOV     #-1,@#TEMP
2294  005672  042737  125252  015122              BIC     #125252,@#TEMP
2295  005700  022737  052525  015122              CMP     #052525,@#TEMP
2296  005706  001401                              BEQ     .+4
2297  005710  104000                              HLT                     ;*ERROR* BIC FAILED
2298  005712  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2299
2300
2301
2302  005714  012737  177777  015112              MOV     #-1,@#C
2303  005722  043737  015070  015112              BIC     @#B,@#C
2304  005730  023727  015112  052525              CMP     @#C,#52525
2305  005736  001401                              BEQ     .+4
2306  005740  104000                              HLT                     ;*ERROR* BIC FAILED
2307  005742  104400                              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2308
2309
2310
```

```
                                ;*****************************************************************
                                ;TEST SUBTRACT INSTRUCTION
                                ;*****************************************************************

005744  012700  125252          MOV     #125252,R0
005750  163700  015070          SUB     @#B,R0
005754  020027  000000          CMP     R0,#0
005760  001401                  BEQ     .+4
005762  104000                  HLT                     ;*ERROR* SUB FAILED
005764  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


005766  012737  125252  015122  MOV     #125252,@#TEMP
005774  166737  007070  015122  SUB     B,@#TEMP
006002  001401                  BEQ     .+4
006004  104000                  HLT                     ;*ERROR* SUB FAILED
006006  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


006010  012767  125252  007104  MOV     #125252,TEMP
006016  163767  015070  007076  SUB     @#B,TEMP
006024  005767  007072          TST     TEMP
006030  001401                  BEQ     .+4
006032  104000                  HLT                     ;*ERROR* SUB FAILED
006034  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


                                ;*****************************************************************
                                ;TEST UNARYS INDIRECT
                                ;*****************************************************************

006036  012737  177777  015122  MOV     #-1,@#TEMP
006044  005037  015122          CLR     @#TEMP
006050  005737  015122          TST     @#TEMP
006054  001401                  BEQ     .+4
006056  104000                  HLT                     ;*ERROR* TST FAILED
006060  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


006062  012737  125252  015122  MOV     #125252,@#TEMP
006070  005137  015122          COM     @#TEMP
006074  022737  052525  015122  CMP     #052525,@#TEMP
006102  001401                  BEQ     .+4
006104  104000                  HLT                     ;*ERROR* COM FAILED
006106  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


006110  005037  015122          CLR     @#TEMP
006114  005237  015122          INC     @#TEMP
006120  022737  000001  015122  CMP     #1,@#TEMP
006126  001401                  BEQ     .+4
006130  104000                  HLT                     ;*ERROR* INC FAILED
```

```
006132  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

006134  005037  015122                  CLR     @#TEMP
006140  005377  006760                  DEC     @TEMP+2
006144  023727  015122  177777          CMP     @#TEMP,#-1
006152  001401                          BEQ     .+4
006154  104000                          HLT                     ;*ERROR* DEC FAILED
006156  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

006160  012737  000001  015122          MOV     #1,@#TEMP
006166  005437  015122                  NEG     @#TEMP
006172  022737  177777  015122          CMP     #-1,@#TEMP
006200  001401                          BEQ     .+4
006202  104000                          HLT                     ;*ERROR* NEG FAILED
006204  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

                        ;*****************************************************************
                        ;TEST INDIRECT ADDRESSING WITH INDEXING
                        ;*****************************************************************

                        ;TEST COMPARE INSTRUCTION
006206  027727  006660  125252                  CMP     @B+2,#125252
006214  001401                                  BEQ     .+4
006216  104000                                  HLT                     ;*ERROR* CMP FAILED
006220  104400                                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

006222  022777  125252  006642                  CMP     #125252,@B+2
006230  001401                                  BEQ     .+4
006232  104000                                  HLT                     ;*ERROR* CMP FAILED
006234  104400                                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

006236  027777  006630  006626                  CMP     @B+2,@B+2
006244  001401                                  BEQ     .+4
006246  104000                                  HLT                     ;*ERROR* CMP FAILED
006250  104400                                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

                        ;*****************************************************************
                        ;TEST MOVE INSTRUCTIONS
                        ;*****************************************************************

006252  017700  006614                  MOV     @B+2,R0
006256  022700  125252                  CMP     #125252,R0
006262  001401                          BEQ     .+4
006264  104000                          HLT                     ;*ERROR* MOV FAILED
```

```
2423  006266  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2424
2425
2426
2427  006270  012777  125252  006626          MOV     #125252,@TEMP+2
2428  006276  023737  015070  015122          CMP     @#B,@#TEMP
2429  006304  001401                          BEQ     .+4
2430  006306  104000                          HLT                     ;*ERROR* MOV FAILED
2431  006310  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2432
2433
2434
2435  006312  017777  006554  006574          MOV     @B+2,@C+2
2436  006320  023737  015070  015112          CMP     @#B,@#C
2437  006326  001401                          BEQ     .+4
2438  006330  104000                          HLT
2439  006332  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2440
2441
2442
2443                                  ;*************************************************************************
2444                                  ;TEST BIC INSTRUCTION INDIRECT WITH INDEXING
2445                                  ;*************************************************************************
2446
2447  006334  012700  177777                  MOV     #-1,R0
2448  006340  047700  006526                  BIC     @B+2,R0
2449  006344  020027  052525                  CMP     R0,#52525
2450  006350  001401                          BEQ     .+4
2451  006352  104000                          HLT                     ;*ERROR* BIC FAILED
2452  006354  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2453
2454
2455
2456  006356  012737  177777  015122          MOV     #-1,@#TEMP
2457  006364  042777  125252  006532          BIC     #125252,@TEMP+2
2458  006372  022737  052525  015122          CMP     #52525,@#TEMP
2459  006400  001401                          BEQ     .+4
2460  006402  104000                          HLT                     ;*ERROR* BIC FAILED
2461  006404  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2462
2463
2464
2465  006406  012737  177777  015112          MOV     #-1,@#C
2466  006414  047777  006452  006472          BIC     @B+2,@C+2
2467  006422  026737  006462  015112          CMP     A+10,@#C
2468  006430  001401                          BEQ     .+4
2469  006432  104000                          HLT                     ;*ERROR* BIC FAILED
2470  006434  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2471
2472
2473
2474  006436  012700  125252                  MOV     #125252,R0
2475  006442  167700  006424                  SUB     @B+2,R0
2476  006446  020027  000000                  CMP     R0,#0
2477  006452  001401                          BEQ     .+4
2478  006454  104000                          HLT                     ;*ERROR* SUB FAILED
```

```
2473  006456  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2480
2481
2482
2483  006460  012737  125252  015122          MOV     #125252,@#TEMP
2484  006466  166777  006376  006430          SUB     B,@TEMP+2
2485  006474  001401                          BEQ     .+4
2486  006476  104000                          HLT                     ;*ERROR* SUB FAILED
2487  006500  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2488
2489
2490
2491  006502  012737  125252  015122          MOV     #125252,@#TEMP
2492  006510  167777  006356  006406          SUB     @B+2,@TEMP+2
2493  006516  005737  015122                  TST     @#TEMP
2494  006522  001401                          BEQ     .+4
2495  006524  104000                          HLT                     ;*ERROR* SUB FAILED
2496  006526  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2497
2498
2499
2500                                  ;*************************************************************************
2501                                  ;TEST ADD INDIRECT WITH INDEXING
2502                                  ;*************************************************************************
2503
2504  006530  005000                          CLR     R0
2505  006532  067700  006334                  ADD     @B+2,R0
2506  006536  022700  125252                  CMP     #125252,R0
2507  006542  001401                          BEQ     .+4
2508  006544  104000                          HLT                     ;*ERROR* ADD FAILED
2509  006546  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2510
2511
2512
2513  006550  005037  015122                  CLR     @#TEMP
2514  006554  062777  125252  006342          ADD     #125252,@TEMP+2
2515  006562  022737  125252  015122          CMP     #125252,@#TEMP
2516  006570  001401                          BEQ     .+4
2517  006572  104000                          HLT                     ;*ERROR* ADD FAILED
2518  006574  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2519
2520
2521  006576  012737  125252  015122          MOV     #125252,@#TEMP
2522  006604  067777  006276  006312          ADD     @A+6,@TEMP+2
2523  006612  023727  015122  177777          CMP     @#TEMP,#-1
2524  006620  001401                          BEQ     .+4
2525  006622  104000                          HLT                     ;*ERROR* ADD FAILED
2526  006624  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2527
2528
2529
2530                                  ;*************************************************************************
2531                                  ;TEST UNARYS INDIRECT WITH INDEXING
2532                                  ;*************************************************************************
2533
2534  006626  012737  177777  015122          MOV     #-1,@#TEMP
```

```
2535  006634  005077  006264          CLR     @TEMP+2
2536  006640  005737  015122          TST     @#TEMP
2537  006644  001401                  BEQ     .+4
2538  006646  104000                  HLT                     ;*ERROR* TST FAILED
2539  006650  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2540
2541
2542
2543  006652  012737  125252  015122  MOV     #125252,@#TEMP
2544  006660  005177  006240          COM     @TEMP+2
2545  006664  022737  052525  015122  CMP     #052525,@#TEMP
2546  006672  001401                  BEQ     .+4
2547  006674  104000                  HLT                     ;*ERROR* COM FAILED
2548  006676  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2549
2550
2551
2552  006700  005037  015122          CLR     @#TEMP
2553  006704  005277  006214          INC     @TEMP+2
2554  006710  022737  000001  015122  CMP     #1,@#TEMP
2555  006716  001401                  BEQ     .+4
2556  006720  104000                  HLT                     ;*ERROR* INC FAILED
2557  006722  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2558
2559
2560
2561  006724  005037  015122          CLR     @#TEMP
2562  006730  005377  006170          DEC     @TEMP+2
2563  006734  023727  015122  177777  CMP     @#TEMP,#-1
2564  006742  001401                  BEQ     .+4
2565  006744  104000                  HLT                     ;*ERROR* DEC FAILED
2566  006746  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2567
2568
2569
2570  006750  012737  000001  015122  MOV     #1,@#TEMP
2571  006756  005477  006142          NEG     @TEMP+2
2572  006762  022737  177777  015122  CMP     #-1,@#TEMP
2573  006770  001401                  BEQ     .+4
2574  006772  104000                  HLT                     ;*ERROR* NEG FAILED
2575  006774  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2576
2577
2578
2579  006776  012737  177777  015122  MOV     #-1,@#TEMP
2580  007004  000261                  SEC
2581  007006  005577  006112          ADC     @TEMP+2
2582  007012  005737  015122          TST     @#TEMP
2583  007016  001401                  BEQ     .+4
2584  007020  104000                  HLT                     ;*ERROR* ADC FAILED
2585  007022  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2586
2587
2588
2589  007024  012737  000001  015122  MOV     #1,@#TEMP
2590  007032  000261                  SEC
```

```
2591  007034  005677  006064          SBC     @TEMP+2
2592  007040  005737  015122          TST     @#TEMP
2593  007044  001401                  BEQ     .+4
2594  007046  104000                  HLT                     ;*ERROR* SBC FAILED
2595  007050  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2596
2597
2598
2599                          ;*************************************************************************
2600                          ;TEST OF COMBINED INDEXING AND INDIRECT
2601                          ;*************************************************************************
2602
2603  007052  012700  177772          MOV     #-6,R0
2604  007056  027027  015100  125252  CMP     @A(0),#125252
2605  007064  001401                  BEQ     .+4
2606  007066  104000                  HLT                     ;*ERROR* CMP FAILED
2607  007070  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2608
2609
2610
2611  007072  012700  177772          MOV     #-6,R0
2612  007076  022770  125252  015100  CMP     #125252,@A(0)
2613  007104  001401                  BEQ     .+4
2614  007106  104000                  HLT                     ;*ERROR* CMP FAILED
2615  007110  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2616
2617
2618
2619  007112  012700  177772          MOV     #-6,R0
2620  007116  012701  000002          MOV     #+2,R1
2621  007122  027071  015100  015100  CMP     @A(0),@A(1)
2622  007130  001401                  BEQ     .+4
2623  007132  104000                  HLT                     ;*ERROR* CMP FAILED
2624  007134  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2625
2626
2627
2628                          ;*************************************************************************
2629                          ;TEST BIC INSTRUCTION
2630                          ;*************************************************************************
2631
2632  007136  012700  000006          MOV     #+6,R0
2633  007142  012767  177777  005752  MOV     #-1,TEMP
2634  007150  047067  015100  005744  BIC     @A(0),TEMP
2635  007156  022767  125252  005736  CMP     #125252,TEMP
2636  007164  001401                  BEQ     .+4
2637  007166  104000                  HLT                     ;*ERROR* BIC FAILED
2638  007170  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2639
2640
2641
2642  007172  012700  177772          MOV     #-6,R0
2643  007176  012737  177777  015112  MOV     #-1,@#C
2644  007204  042770  125252  015122  BIC     #125252,@TEMP(0)
2645  007212  023727  015112  052525  CMP     @#C,#052525
2646  007220  001401                  BEQ     .+4
```

```
2647  007222  104000                    HLT                     ;*ERROR* BIC FAILED
2648  007224  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2649
2650
2651  007226  012737  177777  015112    MOV     #-1,@#C
2652  007234  012700  177772            MOV     #-6,R0
2653  007240  012701  177772            MOV     #-6,R1
2654  007244  047071  015100  015122    BIC     @A(0),@TEMP(1)
2655  007252  022737  052525  015112    CMP     #052525,@#C
2656  007260  001401                    BEQ     .+4
2657  007262  104000                    HLT                     ;*ERROR* BIC FAILED
2658  007264  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2659
2660
2661
2662  007266  122727  000000  000001    CMPB    #0,#1
2663  007274  002401                    BLT     .+4
2664  007276  104000                    HLT                     ;*ERROR* CMPB FAILED
2665  007300  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2666
2667
2668                          ;*************************************************************************
2669                          ;TEST COMPARE INSTRUCTION INDEXED
2670                          ;*************************************************************************
2671
2672  007302  012700  177770            MOV     #-10,R0         ;MINUS 10 TO REG 0
2673  007306  126027  015100  000252    CMPB    A(0),#000252    ;(A INDEX BY MINUS 10) TO #125252
2674  007314  001401                    BEQ     .+4
2675  007316  104000                    HLT                     ;*ERROR* COMPARE WITH INDEX FAILED
2676  007320  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2677
2678
2679
2680  007322  012700  177770            MOV     #-10,R0         ;FOR INDEX
2681  007326  122760  000252  015100    CMPB    #000252,A(0)    ;A INDEXED
2682  007334  001401                    BEQ     .+4
2683  007336  104000                    HLT                     ;*ERROR* CMPB FAILED
2684  007340  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2685
2686
2687
2688  007342  012700  000010            MOV     #10,R0          ;INDEX
2689  007346  126027  015100  000125    CMPB    A(0),#000125
2690  007354  001401                    BEQ     .+4
2691  007356  104000                    HLT                     ;*ERROR* CMPB FAILED
2692  007360  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2693
2694
2695
2696  007362  012700  000010            MOV     #10,R0
2697  007366  122760  000125  015100    CMPB    #000125,A(0)
2698  007374  001401                    BEQ     .+4
2699  007376  104000                    HLT                     ;*ERROR* CMPB FAILED
2700  007400  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2701
2702
```

```
2703
2704  007402  012700  177770            MOV     #-10,R0
2705  007406  126060  015100  015100    CMPB    A(0),A(0)
2706  007414  001401                    BEQ     .+4
2707  007416  104000                    HLT                     ;*ERROR* CMPB FAILED
2708  007420  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2709
2710
2711
2712  007422  012700  000010            MOV     #+10,R0
2713  007426  126060  015100  015100    CMPB    A(0),A(0)
2714  007434  001401                    BEQ     .+4
2715  007436  104000                    HLT                     ;*ERROR* CMPB FAILED
2716  007440  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2717
2718
2719
2720  007442  012700  177770            MOV     #-10,R0
2721  007446  012701  000004            MOV     #+4,R1
2722  007452  126061  015100  015100    CMPB    A(0),A(1)
2723  007460  001401                    BEQ     .+4
2724  007462  104000                    HLT                     ;*ERROR* CMPB FAILED
2725  007464  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2726
2727
2728
2729  007466  126160  015100  015100    CMPB    A(1),A(0)
2730  007474  001401                    BEQ     .+4
2731  007476  104000                    HLT                     ;*ERROR* CMPB FAILED
2732  007500  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2733
2734
2735
2736  007502  012700  177774            MOV     #-4,R0
2737  007506  012701  000010            MOV     #+10,R1
2738  007512  126061  015100  015100    CMPB    A(0),A(1)
```

```
2741  007520  001401                          BEQ     .+4
2742  007522  104000                          HLT                     ;*ERROR* CMPB FAILED
2743  007524  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2744
2745
2746
2747  007526  012700  177774                  MOV     #-4,R0
2748  007532  012701  000010                  MOV     #10,R1
2749  007536  126160  015100  015100          CMPB    A(1),A(0)
2750  007544  001401                          BEQ     .+4
2751  007546  104000                          HLT                     ;*ERROR* CMPB FAILED
2752  007550  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2753
2754
2755                                  ;*********************************************************************
2756                                  ;TEST MOVE INSTRUCTION FOR INDEX
2757                                  ;*********************************************************************
2758
2759
2760  007552  012700  177770                  MOV     #-10,R0
2761  007556  116067  015100  005336          MOVB    A(0),TEMP
2762  007564  126727  005332  000252          CMPB    TEMP,#000252
2763  007572  001401                          BEQ     .+4
2764  007574  104000                          HLT                     ;*ERROR* MOVB FAILED
2765  007576  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2766
2767
2768
2769  007600  012700  000010                  MOV     #+10,R0
2770  007604  116067  015100  005310          MOVB    A(0),TEMP
2771  007612  126727  005304  000125          CMPB    TEMP,#000125
2772  007620  001401                          BEQ     .+4
2773  007622  104000                          HLT                     ;*ERROR* MOVB FAILED
2774  007624  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2775
2776
2777
2778  007626  012700  177770                  MOV     #-10,R0
2779  007632  112760  125252  015122          MOVB    #125252,TEMP(0)
2780  007640  123727  015112  125252          CMPB    @#C,#125252
2781  007646  001401                          BEQ     .+4
2782  007650  104000                          HLT                     ;*ERROR* MOVB FAILED
2783  007652  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2784
2785
2786
2787  007654  012700  000010                  MOV     #+10,R0
2788  007660  112760  052525  015122          MOVB    #052525,TEMP(0)
2789  007666  123727  015132  052525          CMPB    @#TEMP+10,#052525
2790  007674  001401                          BEQ     .+4
2791  007676  104000                          HLT                     ;*ERROR* MOVB FAILED
2792  007700  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
```

```
                                ;*****************************************************************
                                ;TEST BIC INSTRUCTION FOR INDEXING
                                ;*****************************************************************

007702  012767  177777  005212          MOV     #-1,TEMP
007710  012700  177770                  MOV     #-10,R0
007714  146067  015100  005200          BICB    A(0),TEMP
007722  126727  005174  177525          CMPB    TEMP,#177525
007730  001401                          BEQ     .+4
007732  104000                          HLT                     ;*ERROR* BICB FAILED
007734  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


007736  012767  177777  005156          MOV     #-1,TEMP
007744  012700  000010                  MOV     #10,R0
007750  146067  015100  005144          BICB    A(0),TEMP
007756  126727  005140  007652          CMPB    TEMP,#007652
007764  001401                          BEQ     .+4
007766  104000                          HLT                     ;*ERROR* BICB FAILED
007770  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


007772  012737  177777  015132          MOV     #-1,@#TEMP+10
010000  012700  000010                  MOV     #10,R0
010004  142760  125252  015122          BICB    #125252,TEMP(0)
010012  123727  015132  002525          CMPB    @#TEMP+10,#2525
010020  001401                          BEQ     .+4
010022  104000                          HLT                     ;*ERROR* BICB FAILED
010024  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


010026  012700  177770                  MOV     #-10,R0
010032  012767  177777  005052          MOV     #-1,TEMP-10
010040  142767  052525  005044          BICB    #052525,TEMP-10
010046  126727  005040  125252          CMPB    TEMP-10,#125252
010054  001401                          BEQ     .+4
010056  104000                          HLT                     ;*ERROR* BICB FAILED
010060  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


                                ;*****************************************************************
                                ;TEST UNARYS INDEXED
                                ;*****************************************************************

010062  012737  177777  015122          MOV     #-1,@#TEMP
010070  012700  177770                  MOV     #-10,R0
010074  105060  015132                  CLRB    D(0)
010100  105737  015122                  TSTB    @#TEMP
010104  001401                          BEQ     .+4
010106  104000                          HLT                     ;*ERROR* CLRB FAILED
010110  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
```

```
2850
2851
2852  010112  012737  177777  015122          MOV     #-1,@#TEMP
2853  010120  012700  177770                  MOV     #-10,R0
2854  010124  105060  015132                  CLRB    D(0)
2855  010130  023727  015122  177400          CMP     @#TEMP,#177400
2856  010136  001401                          BEQ     .+4
2857  010140  104000                          HLT                     ;*ERROR* CLRB FAILED
2858  010142  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2859
2860
2861
2862  010144  012737  177777  015122          MOV     #-1,@#TEMP
2863  010152  012700  177771                  MOV     #-7,R0
2864  010156  105060  015132                  CLRB    D(0)
2865  010162  023727  015122  000377          CMP     @#TEMP,#000377
2866  010170  001401                          BEQ     .+4
2867  010172  104000                          HLT                     ;*ERROR* CLRB FAILED
2868  010174  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2869
2870
2871
2872  010176  012737  177777  015122          MOV     #-1,@#TEMP
2873  010204  012700  000010                  MOV     #+10,R0
2874  010210  105060  015112                  CLRB    C(0)
2875  010214  105737  015122                  TSTB    @#TEMP
2876  010220  001401                          BEQ     .+4
2877  010222  104000                          HLT                     ;*ERROR* CLRB FAILED
2878  010224  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2879
2880
2881
2882  010226  012737  177777  015122          MOV     #-1,@#TEMP
2883  010234  012700  177770                  MOV     #-10,R0
2884  010240  105160  015132                  COMB    D(0)
2885  010244  105737  015122                  TSTB    @#TEMP
2886  010250  001401                          BEQ     .+4
2887  010252  104000                          HLT                     ;*ERROR* COMB FAILED
2888  010254  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2889
2890
2891
2892  010256  012737  177777  015122          MOV     #-1,@#TEMP
2893  010264  012700  000010                  MOV     #10,R0
2894  010270  105160  015112                  COMB    C(0)
2895  010274  105737  015122                  TSTB    @#TEMP
2896  010300  001401                          BEQ     .+4
2897  010302  104000                          HLT                     ;*ERROR* COMB FAILED
2898  010304  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2899
2900
2901  010306  012737  177777  015122          MOV     #-1,@#TEMP
2902  010314  012700  177770                  MOV     #-10,R0
2903  010320  105260  015132                  INCB    D(0)
2904  010324  105737  015122                  TSTB    @#TEMP
2905  010330  001401                          BEQ     .+4
```

```
2906  010332  104000                    HLT                     ;*ERROR* INCB FAILED
2907  010334  023727  015122  177400    CMP     @#TEMP,#177400
2908  010342  001401                    BEQ     .+4
2909  010344  104000                    HLT                     ;*ERROR* INCB FAILED
2910  010346  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2911
2912
2913
2914  010350  012737  177777  015122    MOV     #-1,@#TEMP
2915  010356  012700  000010            MOV     #+10,R0
2916  010362  105260  015112            INCB    C(0)
2917  010366  105737  015122            TSTB    @#TEMP
2918  010372  001401                    BEQ     .+4
2919  010374  104000                    HLT                     ;*ERROR* INCB FAILED
2920  010376  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2921
2922
2923
2924  010400  012737  000001  015122    MOV     #1,@#TEMP
2925  010406  012700  177770            MOV     #-10,R0
2926  010412  105360  015132            DECB    D(0)
2927  010416  105737  015122            TSTB    @#TEMP
2928  010422  001401                    BEQ     .+4
2929  010424  104000                    HLT                     ;*ERROR* DECB FAILED
2930  010426  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2931
2932
2933
2934  010430  012737  000001  015122    MOV     #1,@#TEMP
2935  010436  012700  000010            MOV     #10,R0
2936  010442  105360  015112            DECB    C(0)
2937  010446  105737  015122            TSTB    @#TEMP
2938  010452  001401                    BEQ     .+4
2939  010454  104000                    HLT                     ;*ERROR* DECB FAILED
2940  010456  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2941
2942
2943
2944  010460  012737  000001  015122    MOV     #1,@#TEMP
2945  010466  012700  177770            MOV     #-10,R0
2946  010472  105460  015132            NEGB    D(0)
2947  010476  023727  015122  000377    CMP     @#TEMP,#377
2948  010504  001401                    BEQ     .+4
2949  010506  104000                    HLT                     ;*ERROR* NEGB FAILED
2950  010510  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2951
2952
2953
2954  010512  012737  000001  015122    MOV     #1,@#TEMP
2955  010520  012700  000010            MOV     #+10,R0
2956  010524  105460  015112            NEGB    C(0)
2957  010530  023727  015122  000377    CMP     @#TEMP,#377
2958  010536  001401                    BEQ     .+4
2959  010540  104000                    HLT                     ;*ERROR* NEGB FAILED
2960  010542  104400                    SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2961
```

```
2962
2963
2964  010544  012737  177777  015122        MOV     #-1,@#TEMP
2965  010552  012700  177770                MOV     #-10,R0
2966  010556  000261                        SEC
2967  010560  105560  015132                ADCB    D(0)
2968  010564  023727  015122  177400        CMP     @#TEMP,#177400
2969  010572  001401                        BEQ     .+4
2970  010574  104000                        HLT                     ;*ERROR* ADCB FAILED
2971  010576  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2972
2973
2974
2975  010600  012737  177777  015122        MOV     #-1,@#TEMP
2976  010606  012700  000010                MOV     #+10,R0
2977  010612  000261                        SEC
2978  010614  105560  015112                ADCB    C(0)
2979  010620  023727  015122  177400        CMP     @#TEMP,#177400
2980  010626  001401                        BEQ     .+4
2981  010630  104000                        HLT                     ;*ERROR* ADCB FAILED
2982  010632  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2983
2984
2985
2986  010634  012737  000401  015122        MOV     #401,@#TEMP
2987  010642  012700  177771                MOV     #-7,R0
2988  010646  000261                        SEC
2989  010650  105660  015132                SBCB    D(0)
2990  010654  022737  000001  015122        CMP     #1,@#TEMP
2991  010662  001401                        BEQ     .+4
2992  010664  104000                        HLT                     ;*ERROR* SBCB FAILED
2993  010666  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
2994
2995
2996
2997  010670  012737  000001  015122        MOV     #1,@#TEMP
2998  010676  012700  000010                MOV     #+10,R0
2999  010702  000261                        SEC
3000  010704  105660  015112                SBCB    C(0)
3001  010710  005737  015122                TST     @#TEMP
3002  010714  001401                        BEQ     .+4
3003  010716  104000                        HLT                     ;*ERROR* SBCB FAILED
3004  010720  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3005
3006
3007
```

```
                                ;*****************************************************************
                                ;TEST INDIRECT ADDRESSING
                                ;*****************************************************************

                                ;TEST COMPARE INSTRUCTION
010722  123727  015070  000252          CMPB    @#B,#000252
010730  001401                          BEQ     .+4
010732  104000                          HLT                     ;*ERROR* CMPB FAILED
010734  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

010736  123727  015071  000252          CMPB    @#B+1,#252
010744  001401                          BEQ     .+4
010746  104000                          HLT                     ;*ERROR* CMPB FAILED
010750  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

010752  122737  125252  015070          CMPB    #125252,@#B
010760  001401                          BEQ     .+4
010762  104000                          HLT                     ;*ERROR* CMPB FAILED
010764  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

010766  123737  015070  015070          CMPB    @#B,@#B
010774  001401                          BEQ     .+4
010776  104000                          HLT                     ;*ERROR* CMPB FAILED
011000  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN


                                ;*****************************************************************
                                ;TEST MOVE INSTRUCTIONS
                                ;*****************************************************************

011002  113700  015070                  MOVB    @#B,R0
011006  122700  000252                  CMPB    #000252,R0
011012  001401                          BEQ     .+4
011014  104000                          HLT                     ;*ERROR* MOVB FAILED
011016  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

011020  112737  125252  015122          MOVB    #125252,@#TEMP
011026  126737  004036  015122          CMPB    B,@#TEMP
011034  001401                          BEQ     .+4
011036  104000                          HLT                     ;*ERROR* MOVB FAILED
011040  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

011042  113737  015070  015112          MOVB    @#B,@#C
011050  126737  004014  015112          CMPB    B,@#C
```

```
3065  011056  001401                        BEQ     .+4
3066  011060  104000                        HLT                     ;*ERROR* MOVB FAILED
3067  011062  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3068
3069
3070                                ;*************************************************************************
3071                                ;TEST UNARYS INDIRECT
3072                                ;*************************************************************************
3073
3074  011064  012737  177777  015122        MOV     #-1,@#TEMP
3075  011072  105037  015122                CLRB    @#TEMP
3076  011076  023727  015122  177400        CMP     @#TEMP,#177400
3077  011104  001401                        BEQ     .+4
3078  011106  104000                        HLT                     ;*ERROR* CLRB FAILED
3079  011110  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3080
3081
3082
3083  011112  012737  125252  015122        MOV     #125252,@#TEMP
3084  011120  105137  015122                COMB    @#TEMP
3085  011124  022737  125125  015122        CMP     #125125,@#TEMP
3086  011132  001401                        BEQ     .+4
3087  011134  104000                        HLT                     ;*ERROR* COMB FAILED
3088  011136  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3089
3090
3091
3092  011140  012737  125252  015122        MOV     #125252,@#TEMP
3093  011146  105137  015123                COMB    @#TEMP+1
3094  011152  022737  052652  015122        CMP     #052652,@#TEMP
3095  011160  001401                        BEQ     .+4
3096  011162  104000                        HLT                     ;*ERROR* COMB FAILED
3097  011164  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3098
3099
3100
3101  011166  005037  015122                CLR     @#TEMP
3102  011172  105237  015123                INCB    @#TEMP+1
3103  011176  022737  000400  015122        CMP     #400,@#TEMP
3104  011204  001401                        BEQ     .+4
3105  011206  104000                        HLT                     ;*ERROR* INCB FAILED
3106  011210  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3107
3108
3109
3110  011212  005037  015122                CLR     @#TEMP
3111  011216  105377  003702                DECB    @TEMP+2
3112  011222  023727  015122  000377        CMP     @#TEMP,#377
3113  011230  001401                        BEQ     .+4
3114  011232  104000                        HLT                     ;*ERROR* DECB FAILED
3115  011234  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3116
3117
3118
3119  011236  005037  015122                CLR     @#TEMP
3120  011242  112737  000001  015123        MOVB    #1,@#TEMP+1
```

```
3121  011250  105437  015123          NEGB    @#TEMP+1
3122  011254  022737  177400  015122  CMP     #177400,@#TEMP
3123  011262  001401                  BEQ     .+4
3124  011264  104000                  HLT                     ;*ERROR* NEGB FAILED
3125  011266  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3126
3127
3128
3129                          ;*************************************************************************
3130                          ;TEST INDIRECT ADDRESSING WITH INDEXING
3131                          ;*************************************************************************
3132
3133                          ;TEST COMPARE INSTRUCTION
3134  011270  127727  003576  125252      CMPB    @B+2,#125252
3135  011276  001401                      BEQ     .+4
3136  011300  104000                      HLT                     ;*ERROR* CMPB FAILED
3137  011302  104400                      SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3138
3139
3140
3141  011304  122777  125252  003560      CMPB    #125252,@B+2
3142  011312  001401                      BEQ     .+4
3143  011314  104000                      HLT                     ;*ERROR* CMPB FAILED
3144  011316  104400                      SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3145
3146
3147
3148  011320  127777  003546  003544      CMPB    @B+2,@B+2
3149  011326  001401                      BEQ     .+4
3150  011330  104000                      HLT                     ;*ERROR* CMPB FAILED
3151  011332  104400                      SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3152
3153
3154                          ;*************************************************************************
3155                          ;TEST MOVE INSTRUCTIONS
3156                          ;*************************************************************************
3157
3158  011334  117700  003532      MOVB    @B+2,R0
3159  011340  122700  125252      CMPB    #125252,R0
3160  011344  001401              BEQ     .+4
3161  011346  104000              HLT                     ;*ERROR* MOVB FAILED
3162  011350  104400              SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3163
3164
3165
3166  011352  112777  125252  003544      MOVB    #125252,@TEMP+2
3167  011360  126737  003504  015122      CMPB    B,@#TEMP
3168  011366  001401                      BEQ     .+4
3169  011370  104000                      HLT                     ;*ERROR* MOVB FAILED
3170  011372  104400                      SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3171
3172
3173
3174  011374  117777  003472  003512      MOVB    @B+2,@C+2
3175  011402  126737  003462  015112      CMPB    B,@#C
3176  011410  001401                      BEQ     .+4
```

```
3177  011412  104000                    HLT                 ;*ERROR* MOVB FAILED
3178  011414  104400                    SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3179
3180
3181
3182                          ;*********************************************************************
3183                          ;TEST BIC INSTRUCTION INDIRECT WITH INDEXING
3184                          ;*********************************************************************
3185
3186  011416  012700  177777          MOV     #-1,R0
3187  011422  147700  003444          BICB    @B+2,R0
3188  011426  120027  052525          CMPB    R0,#52525
3189  011432  001401                  BEQ     .+4
3190  011434  104000                  HLT                 ;*ERROR* BICB FAILED
3191  011436  104400                  SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3192
3193
3194
3195  011440  012737  177777  015122  MOV     #-1,@#TEMP
3196  011446  142777  125252  003450  BICB    #125252,@TEMP+2
3197  011454  122737  052525  015122  CMPB    #52525,@#TEMP
3198  011462  001401                  BEQ     .+4
3199  011464  104000                  HLT                 ;*ERROR* BICB FAILED
3200  011466  104400                  SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3201
3202
3203
3204  011470  012737  177777  015112  MOV     #-1,@#C
3205  011476  147777  003370  003410  BICB    @B+2,@C+2
3206  011504  126737  003400  015112  CMPB    A+10,@#C
3207  011512  001401                  BEQ     .+4
3208  011514  104000                  HLT                 ;*ERROR* BICB FAILED
3209  011516  104400                  SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3210
3211
3212                          ;*********************************************************************
3213                          ;TEST UNARYS INDIRECT WITH INDEXING
3214                          ;*********************************************************************
3215
3216  011520  012737  177777  015122  MOV     #-1,@#TEMP
3217  011526  105077  003372          CLRB    @TEMP+2
3218  011532  105737  015122          TSTB    @#TEMP
3219  011536  001401                  BEQ     .+4
3220  011540  104000                  HLT                 ;*ERROR* CLRB FAILED
3221  011542  104400                  SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3222
3223
3224
3225  011544  012737  125252  015122  MOV     #125252,@#TEMP
3226  011552  105177  003346          COMB    @TEMP+2
3227  011556  122737  052525  015122  CMPB    #052525,@#TEMP
3228  011564  001401                  BEQ     .+4
3229  011566  104000                  HLT                 ;*ERROR* COMB FAILED
3230  011570  104400                  SCOPE               ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3231
3232
```

```
3233
3234  011572  005037  015122          CLR     @#TEMP
3235  011576  105277  003322          INCB    @TEMP+2
3236  011602  122737  000001  015122  CMPB    #1,@#TEMP
3237  011610  001401                  BEQ     .+4
3238  011612  104000                  HLT                     ;*ERROR* INCB FAILED
3239  011614  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3240
3241
3242
3243  011616  005037  015122          CLR     @#TEMP
3244  011622  105377  003276          DECB    @TEMP+2
3245  011626  123727  015122  177777  CMPB    @#TEMP,#-1
3246  011634  001401                  BEQ     .+4
3247  011636  104000                  HLT                     ;*ERROR* DECB FAILED
3248  011640  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3249
3250
3251
3252  011642  012737  000001  015122  MOV     #1,@#TEMP
3253  011650  105477  003250          NEGB    @TEMP+2
3254  011654  122737  177777  015122  CMPB    #-1,@#TEMP
3255  011662  001401                  BEQ     .+4
3256  011664  104000                  HLT                     ;*ERROR* NEGB FAILED
3257  011666  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3258
3259
3260
3261  011670  012737  177777  015122  MOV     #-1,@#TEMP
3262  011676  000261                  SEC
3263  011700  105577  003220          ADCB    @TEMP+2
3264  011704  022737  177400  015122  CMP     #177400,@#TEMP
3265  011712  001401                  BEQ     .+4
3266  011714  104000                  HLT                     ;*ERROR* ADCB FAILED
3267  011716  105737  015122          TSTB    @#TEMP
3268  011722  001401                  BEQ     .+4
3269  011724  104000                  HLT                     ;*ERROR* TSTB FAILED
3270  011726  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3271
3272
3273
3274  011730  012737  000001  015122  MOV     #1,@#TEMP
3275  011736  000261                  SEC
3276  011740  105377  003160          DECB    @TEMP+2
3277  011744  005737  015122          TST     @#TEMP
3278  011750  001401                  BEQ     .+4
3279  011752  104000                  HLT                     ;*ERROR* DECB FAILED
3280  011754  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3281
3282
3283
3284                          ;*************************************************************************
3285                          ;TEST OF COMBINED INDEXING AND INDIRECT
3286                          ;*************************************************************************
3287
3288  011756  012700  177772          MOV     #-6,R0
```

```
3289  011762  127027  015100  125252          CMPB    @A(0),#125252
3290  011770  001401                          BEQ     .+4
3291  011772  104000                          HLT                     ;*ERROR* CMPB FAILED
3292  011774  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3293
3294
3295
3296  011776  012700  177772                  MOV     #-6,R0
3297  012002  122770  125252  015100          CMPB    #125252,@A(0)
3298  012010  001401                          BEQ     .+4
3299  012012  104000                          HLT                     ;*ERROR* CMPB FAILED
3300  012014  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3301
3302
3303
3304  012016  012700  177772                  MOV     #-6,R0
3305  012022  012701  000002                  MOV     #+2,R1
3306  012026  127071  015100  015100          CMPB    @A(0),@A(1)
3307  012034  001401                          BEQ     .+4
3308  012036  104000                          HLT                     ;*ERROR* CMPB FAILED
3309  012040  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3310
3311
3312                          ;*************************************************************
3313                          ;TEST BIC INSTRUCTION
3314                          ;*************************************************************
3315
3316  012042  012700  000006                  MOV     #+6,R0
3317  012046  012767  177777  003046          MOV     #-1,TEMP
3318  012054  147067  015100  003040          BICB    @A(0),TEMP
3319  012062  122767  125252  003032          CMPB    #125252,TEMP
3320  012070  001401                          BEQ     .+4
3321  012072  104000                          HLT                     ;*ERROR* BICB FAILED
3322  012074  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3323
3324
3325
3326  012076  012700  177772                  MOV     #-6,R0
3327  012102  012737  177777  015112          MOV     #-1,@#C
3328  012110  142770  125252  015122          BICB    #125252,@TEMP(0)
3329  012116  123727  015112  000125          CMPB    @#C,#000125
3330  012124  001401                          BEQ     .+4
3331  012126  104000                          HLT                     ;*ERROR* BICB FAILED
3332  012130  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3333
3334
3335
3336  012132  012700  015072                  MOV     #B+2,R0         ;ADDRESS OF ADDRESS OF B
3337  012136  023067  002726                  CMP     @(0)+,B
3338  012142  001401                          BEQ     .+4
3339  012144  104000                          HLT                     ;*ERROR* CMP FAILED
3340  012146  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3341
3342
3343
3344  012150  012700  015074                  MOV     #B+4,R0
```

```
3345  012154  025067  002710                  CMP     @-(0),B
3346  012160  001401                          BEQ     .+4
3347  012162  104000                          HLT                     ;*ERROR* CMP FAILED
3348  012164  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3349
3350
3351
3352  012166  012700  015074                  MOV     #B+4,R0
3353  012172  125067  002672                  CMPB    @-(0),B
3354  012176  001401                          BEQ     .+4
3355  012200  104000                          HLT                     ;*ERROR* CMPB FAILED
3356  012202  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3357
3358
3359
3360  012204  012700  015116                  MOV     #C+4,R0
3361  012210  012737  177777  015112          MOV     #-1,@#C
3362  012216  105050                          CLRB    @-(0)
3363  012220  023727  015112  177400          CMP     @#C,#177400
3364  012226  001401                          BEQ     .+4
3365  012230  104000                          HLT                     ;*ERROR* CLRB FAILED
3366  012232  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3367
3368
3369  012234  012737  177777  015112          MOV     #-1,@#C
3370  012242  012700  177772                  MOV     #-6,R0
3371  012246  012701  177772                  MOV     #-6,R1
3372  012252  147071  015100  015122          BICB    @A(0),@TEMP(1)
3373  012260  022737  177525  015112          CMP     #177525,@#C
3374  012266  001401                          BEQ     .+4
3375  012270  104000                          HLT                     ;*ERROR* BICB FAILED
3376  012272  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3377
3378
3379                          ;*************************************************************************
3380                          ;TEST JSR INSTRUCTION
3381                          ;*************************************************************************
3382
3383
3384  012274  004767  000002          JSR     PC, TJSR2       ;PLACE PC ON STACK
3385  012300  000405          TJSR1:  BR      TJSR3           ;RETURN HERE ON RTS PC
3386  012302  121627  012300  TJSR2: CMPB    @SP,#TJSR1      ;CHECK FOR CORRECT PC ON STACK
3387  012306  001401                  BEQ     .+4
3388  012310  104000                  HLT                     ;*ERROR* INCORRECT PC ON STACK
3389  012312  000207                  RTS     PC              ;RETURN TO INST AFTER JSR
3390  012314  104400          TJSR3:  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3391
3392
3393
3394  012316  000257                  CCC
3395  012320  004717                  JSR     PC,@PC          ;INSTRUCTION UNDER TEST
3396  012322  121627  012322          CMPB    @SP,#TJSR3+6    ;TEST THE STACK
3397  012326  001401                  BEQ     .+4
3398  012330  104000                  HLT                     ;*ERROR* PC OF JSR DID NOT GO TO STACK
3399  012332  005726                  TST     (6)+            ;REPOSITION THE STACK
3400  012334  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
```

```
3401
3402
3403                                    ;*********************************************************************
3404                                    ;TEST NESTED SUBROUTINES
3405                                    ;*********************************************************************
3406
3407
3408  012336  000257                            CCC                     ;CLEAR CONDITION CODES
3409  012340  004767  002622                   JSR     PC,SUBR6
3410  012344  100401                           BMI     .+4
3411  012346  104000                           HLT                     ;*ERROR* JSR OR RTS FAILED
3412  012350  001401                           BEQ     .+4
3413  012352  104000                           HLT                     ;*ERROR* JSR OR RTS FAILED
3414  012354  102401                           BVS     .+4
3415  012356  104000                           HLT                     ;*ERROR* JSR OR RTS FAILED
3416  012360  103401                           BCS     .+4
3417  012362  104000                           HLT                     ;*ERROR* JSR OR RTS FAILED
3418  012364  104400                           SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3419
3420
3421                                    ;*********************************************************************
3422                                    ;TEST ROTATE ODD BYTE
3423                                    ;*********************************************************************
3424
3425  012366  104400                           SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3426
3427
3428  012370  000257                            CCC                     ;CLEAR "C"
3429  012372  012767  123456  002522           MOV     #123456,TEMP
3430  012400  106067  002517                   RORB    TEMP+1          ;ROTATE ODD BYTE
3431  012404  103401                           BCS     .+4
3432  012406  104000                           HLT                     ;*ERROR* C NOT SET
3433  012410  102401                           BVS     .+4
3434  012412  104000                           HLT                     ;*ERROR* V NOT SET
3435  012414  022767  051456  002500           CMP     #051456,TEMP
3436  012422  001401                           BEQ     .+4
3437  012424  104000                           HLT                     ;*ERROR* ROTATE FAILED
3438  012426  104400                           SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3439
3440
3441  012430  000277                            SCC                     ;SET C
3442  012432  012767  123456  002462           MOV     #123456,TEMP
3443  012440  106067  002457                   RORB    TEMP+1
3444  012444  103401                           BCS     .+4
3445  012446  104000                           HLT                     ;*ERROR* C NOT SET
3446  012450  102001                           BVC     .+4
3447  012452  104000                           HLT                     ;*ERROR* V NOT CLEARED
3448  012454  022767  151456  002440           CMP     #151456,TEMP
3449  012462  001401                           BEQ     .+4
3450  012464  104000                           HLT                     ;*ERROR* ROTATE FAILED
3451  012466  104400                           SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3452
3453
3454
3455  012470  000257                            CCC
3456  012472  012767  123456  002422           MOV     #123456,TEMP
```

```
3457  012500  106167  002417          ROLB    TEMP+1
3458  012504  103401                  BCS     .+4
3459  012506  104000                  HLT                     ;*ERROR* C NOT SET
3460  012510  102401                  BVS     .+4
3461  012512  104000                  HLT                     ;*ERROR* V NOT SET
3462  012514  022767  047056  002400  CMP     #047056,TEMP
3463  012522  001401                  BEQ     .+4
3464  012524  104000                  HLT                     ;*ERROR* ROTATE BYTE FAILED
3465  012526  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3466
3467
3468
3469  012530  000277                  SCC                     ;SET C
3470  012532  012767  123456  002362  MOV     #123456,TEMP
3471  012540  106167  002357          ROLB    TEMP+1
3472  012544  103401                  BCS     .+4
3473  012546  104000                  HLT                     ;*ERROR* C NOT SET
3474  012550  102401                  BVS     .+4
3475  012552  104000                  HLT                     ;*ERROR* V NOT SET
3476  012554  022767  047456  002340  CMP     #047456,TEMP
3477  012562  001401                  BEQ     .+4
3478  012564  104000                  HLT                     ;*ERROR* ROTATE ODD BYTE FAILED
3479  012566  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3480
3481
3482                                  ;*************************************************************
3483                                  ;TEST ASL INSTRUCTION MODE 0
3484                                  ;*************************************************************
3485
3486  012570  012700  125252  ASL0:   MOV     #125252,R0      ;SET R0=125252
3487  012574  000277                  SCC
3488  012576  006300                  ASL     R0              ;SET R0=52524
3489  012600  103003                  BCC     1$              ;BRANCH IF C FAILED
3490  012602  102002                  BVC     1$              ;BRANCH IF V FAILED
3491  012604  001401                  BEQ     1$              ;BRANCH IF Z FAILED
3492  012606  100001                  BPL     .+4             ;BRANCH IF N OK
3493  012610  104000          1$:     HLT                     ;*ERROR* ASL MODE 0 FLAGS FAILED
3494
3495  012612  020027  052524          CMP     R0,#52524       ;IS ASL RESULT OK?
3496  012616  001401                  BEQ     .+4             ;BRANCH IF YES
3497  012620  104000                  HLT                     ;*ERROR* ASL MODE 0 FAILED
3498  012622  104400                  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3499
3500
3501                                  ;*************************************************************
3502                                  ;TEST ASL MODE NON 0 (67)
3503                                  ;*************************************************************
3504
3505  012624  012767  032525  002270  ASL67:  MOV     #32525,TEMP     ;TEMP=32525
3506  012632  000257                  CCC
3507  012634  006367  002262          ASL     TEMP            ;TEMP=65252
3508  012640  103403                  BCS     1$              ;BRANCH IF C FAILED
3509  012642  102402                  BVS     1$              ;BRANCH IF V FAILED
3510  012644  001401                  BEQ     1$              ;BRANCH IF Z FAILED
3511  012646  100001                  BPL     .+4             ;BRANCH IF N OK
3512  012650  104000          1$:     HLT                     ;*ERROR* ASL MODE 67 FLATS FAILED
```

```
3513
3514  012652  026727  002244  065252          CMP     TEMP,#65252     ;IS ASL RESULT OK?
3515  012660  001401                          BEQ     .+4             ;BRANCH IF YES
3516  012662  104000                          HLT                     ;*ERROR* ASL MODE 67 FAILED
3517  012664  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3518
3519
3520                                  ;*************************************************************************
3521                                  ;TEST SWAB INSTRUCTION MODE 0
3522                                  ;*************************************************************************
3523
3524  012666  012700  077777          SWAB0:  MOV     #77777,R0       ;R0=77777
3525  012672  000277                          SCC
3526  012674  000300                          SWAB    R0              ;R0=177577
3527  012676  103403                          BCS     1$              ;BRANCH IF C FAILED
3528  012700  102402                          BVS     1$              ;BRANCH IF V FAILED
3529  012702  001401                          BEQ     1$              ;BRANCH IF Z FAILED
3530  012704  100001                          BPL     .+4             ;BRANCH IF N OK
3531  012706  104000                  1$:     HLT                     ;*ERROR* SWAB MODE 0 FLAGS FAILED
3532
3533  012710  020027  177577                  CMP     R0,#177577      ;IS SWAB RESULT OK?
3534  012714  001401                          BEQ     .+4             ;BRANCH IF YES
3535  012716  104000                          HLT                     ;*ERROR* SWAB MODE 0 FAILED
3536  012720  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3537
3538
3539                                  ;*************************************************************************
3540                                  ;TEST SWAB INSTRUCTION MODE NON 0 (67)
3541                                  ;*************************************************************************
3542
3543  012722  012767  000377  002172  SWAB67: MOV     #377,TEMP       ;TEMP=377
3544  012730  000277                          SCC
3545  012732  000367  002164                  SWAB    TEMP            ;TEMP=177400
3546  012736  103403                          BCS     1$              ;BRANCH IF C FAILED
3547  012740  102402                          BVS     1$              ;BRANCH IF V FAILED
3548  012742  001001                          BNE     1$              ;BRANCH IF Z FAILED
3549  012744  100001                          BPL     .+4             ;BRANCH IF N OK
3550  012746  104000                  1$:     HLT                     ;*ERROR* SWAB MODE 67 FLAGS FAILED
3551
3552  012750  026727  002146  177400          CMP     TEMP,#177400    ;IS SWAB RESULT OK?
3553  012756  001401                          BEQ     .+4             ;BRANCH IF YES
3554  012760  104000                          HLT                     ;*ERROR* SWAB MODE 67 FAILED
3555  012762  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3556
3557                                  ;*************************************************************************
3558                                  ;TEST ASRB AND ASLB INSTRUCTIONS
3559                                  ;*************************************************************************
3560
3561  012764  000257                          CCC                     ;CLEAR C
3562  012766  012767  177777  002126          MOV     #-1,TEMP
3563  012774  106267  002123                  ASRB    TEMP+1
3564  013000  103401                          BCS     .+4
3565  013002  104000                          HLT                     ;*ERROR* C NOT SET
3566  013004  102001                          BVC     .+4
3567  013006  104000                          HLT                     ;*ERROR* V NOT CLEARED
3568  013010  026727  002106  177777          CMP     TEMP,#-1
```

```
3569  013016  001401                          BEQ     .+4
3570  013020  104000                          HLT                     ;*ERROR* SHIFT FAILED
3571  013022  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3572
3573
3574
3575  013024  000277                          SCC
3576  013026  012767  177777  002066          MOV     #-1,TEMP
3577  013034  106367  002063                  ASLB    TEMP+1
3578  013040  103401                          BCS     .+4
3579  013042  104000                          HLT                     ;*ERROR* C NOT SET
3580  013044  102001                          BVC     .+4
3581  013046  104000                          HLT                     ;*ERROR* V NOT CLEARED
3582  013050  026727  002046  177377          CMP     TEMP,#177377
3583  013056  001401                          BEQ     .+4
3584  013060  104000                          HLT                     ;*ERROR* SHIFT BYTE FAILED
3585  013062  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3586
3613  013064  005037  016212                  CLR     @#ICOUNT        ;NO ITERATION
3614
3615                          ;*************************************************************************
3616                          ;TEST ROTATING NUMBERS
3617                          ;*************************************************************************
3618
3619
3620
3621  013070  012767  177777  000142          MOV     #-1,REFF        ;INITIALIZE BASE NUMBER
3622  013076  005267  000136          TSROT:  INC     REFF            ;INCREMENT NUMBER
3623  013102  004767  000012                  JSR     PC,ROTALL       ;GO TO COMPARE ROUTINE
3624  013106  026727  000126  100077          CMP     REFF,#100077    ;TEST ALL VALUES
3625  013114  001370                          BNE     TSROT           ;NO TEST THEM ALL
3626  013116  000452                          BR      TSRT2A          ;WE ARE DONE
3627
3628  013120  016767  000114  000114  ROTALL: MOV     REFF,TEST
3629  013126  006167  000110                  ROL     TEST
3630  013132  006067  000104                  ROR     TEST
3631  013136  006067  000100                  ROR     TEST
3632  013142  006067  000074                  ROR     TEST
3633  013146  006067  000070                  ROR     TEST
3634  013152  006167  000064                  ROL     TEST
3635  013156  006167  000060                  ROL     TEST
3636  013162  006167  000054                  ROL     TEST
3637  013166  100004                          BPL     .+12
  (1) 013170  103007                          BCC     .+20            ;Z=1
  (1) 013172  102013                          BVC     .+30            ;Z=1, C=1
  (1) 013174  104000                          HLT                     ;*ERROR* Z=C, BUT V=1
  (1) 013176  000411                          BR      .+24
  (1) 013200  103006                          BCC     .+16            ;Z=0
  (1) 013202  102407                          BVS     .+20            ;Z=0, C=1
  (1) 013204  104000                          HLT                     ;*ERROR* Z NOT EQUAL C, V=1
  (1) 013206  000405                          BR      .+14
  (1) 013210  102404                          BVS     .+12            ;Z=1, C=0
  (1) 013212  104000                          HLT                     ;*ERROR* Z NOT EQUAL C, V=1
  (1) 013214  000402                          BR      .+6
  (1) 013216  102001                          BVC     .+4             ;Z=0, C=0
  (1) 013220  104000                          HLT                     ;*ERROR* Z=C, BUT V=1
```

```
  (1)  013222  104400                          SCOPE                    ;LOOP ON SUBTEST OR SETUP LOC. RETURN
  (1)
  (1)
3638   013224  026767  000012  000006          CMP     TEST,REFF
3639   013232  001401                          BEQ     .+4
3640   013234  104000                          HLT                      ;*ERROR* INITIAL NOT EQUAL TO FINAL
3641   013236  000207                          RTS     PC               ;ROTATE WORD FAILED
3642   013240  000000                  REFF:   0                        ;GOOD DATA
3643   013242  000000                  TEST:   0                        ;BAD DATA
3644           013240                          REF=REFF
3645                                  ;*************************************************************************
3646                                  ;TEST ROTATING BYTE EVEN/ODD, ALL NUMBERS
3647                                  ;*************************************************************************
3648
3649   013244  012767  177777  177766  TSRT2A: MOV     #-1,REFF
3650   013252  005267  177762          TSROT2: INC     REFF
3651   013256  004767  000016                  JSR     PC,ROTBE
3652   013262  004767  000122                  JSR     PC,ROTBO
3653   013266  022767  177777  177744          CMP     #-1,REFF
3654   013274  001366                          BNE     TSROT2
3655   013276  000505                          BR      ROTEN1
3656   013300  016767  177734  177734  ROTBE:  MOV     REFF,TEST
3657   013306  106067  177730                  RORB    TEST             ;ROTATE BYTE EVEN
3658   013312  106067  177724                  RORB    TEST
3659   013316  106067  177720                  RORB    TEST
3660   013322  106167  177714                  ROLB    TEST
3661   013326  106167  177710                  ROLB    TEST
3662   013332  106167  177704                  ROLB    TEST
3663   013336  100004                          BPL     .+12
  (1)  013340  103007                          BCC     .+20             ;Z=1
  (1)  013342  102013                          BVC     .+30             ;Z=1, C=1
  (1)  013344  104000                          HLT                      ;*ERROR* Z=C, BUT V=1
  (1)  013346  000411                          BR      .+24
  (1)  013350  103006                          BCC     .+16             ;Z=0
  (1)  013352  102407                          BVS     .+20             ;Z=0, C=1
  (1)  013354  104000                          HLT                      ;*ERROR* Z NOT EQUAL C, V=1
  (1)  013356  000405                          BR      .+14
  (1)  013360  102404                          BVS     .+12             ;Z=1, C=0
  (1)  013362  104000                          HLT                      ;*ERROR* Z NOT EQUAL C, V=1
  (1)  013364  000402                          BR      .+6
  (1)  013366  102001                          BVC     .+4              ;Z=0, C=0
  (1)  013370  104000                          HLT                      ;*ERROR* Z=C, BUT V=1
  (1)  013372  104400                          SCOPE                    ;LOOP ON SUBTEST OR SETUP LOC. RETURN
  (1)
  (1)
3664   013374  026767  177642  177636          CMP     TEST,REFF        ;IS RESULT OK?
3665   013402  001401                          BEQ     .+4              ;BRANCH IF YES
3666   013404  104000                          HLT                      ;*ERROR* RORB OR ROLB FAILED
3667   013406  000207                          RTS     PC
3668
3669   013410  106067  177627          ROTBO:  RORB    TEST+1           ;ROTATE BYTE ODD
3670   013414  106067  177623                  RORB    TEST+1
3671   013420  106067  177617                  RORB    TEST+1
3672   013424  106167  177613                  ROLB    TEST+1
3673   013430  106167  177607                  ROLB    TEST+1
3674   013434  106167  177603                  ROLB    TEST+1
```

```
 3675  013440  100004                          BPL     .+12
  (1)  013442  103007                          BCC     .+20            ;Z=1
  (1)  013444  102013                          BVC     .+30            ;Z=1, C=1
  (1)  013446  104000                          HLT                     ;*ERROR* Z=C, BUT V=1
  (1)  013450  000411                          BR      .+24
  (1)  013452  103006                          BCC     .+16            ;Z=0
  (1)  013454  102407                          BVS     .+20            ;Z=0, C=1
  (1)  013456  104000                          HLT                     ;*ERROR* Z NOT EQUAL C, V=1
  (1)  013460  000405                          BR      .+14
  (1)  013462  102404                          BVS     .+12            ;Z=1, C=0
  (1)  013464  104000                          HLT                     ;*ERROR* Z NOT EQUAL C, V=1
  (1)  013466  000402                          BR      .+6
  (1)  013470  102001                          BVC     .+4             ;Z=0, C=0
  (1)  013472  104000                          HLT                     ;*ERROR* Z=C, BUT V=1
  (1)  013474  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
  (1)
  (1)
 3676  013476  026767  177540  177534          CMP     TEST,REFF
 3677  013504  001401                          BEQ     .+4
 3678  013506  104000                          HLT                     ;*ERROR* ROTATE BYTE FAILED
 3679  013510  000207                          RTS     PC
 3680
 3681
 3682
 3683
 3684
 3685  013512  104400              ROTEN1: SCOPE                       ;LOOP ON SUBTEST OR SETUP LOC. RETURN
 3686
 3687
 3688                              ;*********************************************************************
 3689
 3691                              ;TEST MTPS AND MFPS INSTRUCTION MODE 0 ALL 1'S
 3692                              ;*********************************************************************
 3693
 3694                              ;DOES NOT TEST INTERRUPT ENABLE.
 3695
 3696  013514  012700  000377      MTPS0:  MOV     #377,R0             ;SET SOURCE = 377
 3697  013520  106400                      MTPS    R0                  ;TEST MTPS 377
 3698  013522  103003                      BCC     1$                  ;BRANCH IF C BIT FAILED
 3699  013524  102002                      BVC     1$                  ;BRANCH IF V BIT FAILED
 3700  013526  001001                      BNE     1$                  ;BRANCH IF Z BIT FAILED
 3701  013530  100401                      BMI     .+4                 ;BRANCH IF N BIT OK
 3702  013532  104000              1$:     HLT                         ;*ERROR* MTPS MODE 0 1'S FAILED
 3703  013534  106700                      MFPS    R0                  ;TEST MFPS 377 SIGN EXT
 3704  013536  042700  000020              BIC     #20,R0              ;CLR TBIT IF SET
 3705  013542  022700  177757              CMP     #177757,R0          ;RESULT IS SIGN EXT + NO T BIT
 3706  013546  001401                      BEQ     .+4                 ;BRANCH IF RESULT OK.
 3707  013550  104000                      HLT                         ;*ERROR* MFPS MODE 0 1'S FAILED
 3708                                                                  ;SAVR0 CONTAINS ERROR PROCESSOR STATUS
 3709  013552  104400                      SCOPE                       ;LOOP ON SUBTEST OR SETUP LOC. RETURN
 3710
 3729
 3730                              ;*********************************************************************
 3731                              ;TEST MTPS AND MFPS INSTRUCTION MODE NON 0 (MODE 27)
 3732                              ;*********************************************************************
 3733
```

```
3734  013554  106427  000000          MTPS27: MTPS    #0              ;TEST MTPS 0
3735  013560  103403                          BCS     1$              ;BRANCH IF C BIT FAILED
3736  013562  102402                          BVS     1$              ;BRANCH IF V BIT FAILED
3737  013564  001401                          BEQ     1$              ;BRANCH IF Z BIT FAILED
3738  013566  100001                          BPL     .+4             ;BRANCH IF N BIT OK
3739  013570  104000                  1$:     HLT                     ;*ERROR* MTPS MODE 27 FAILED
3740  013572  012767  000377  001322          MOV     #377,TEMP       ;INIT TEMP (MFPS IS A BYTE MOVE)
3741  013600  106767  001316                  MFPS    TEMP            ;TEST MFPS 0
3742  013604  042767  000020  001310          BIC     #20,TEMP        ;CLR TBIT IF SET (DUE TO TBIT PASS)
3743  013612  005767  001304                  TST     TEMP            ;RESULT NO SIGN EXT OR T BIT
3744  013616  001401                          BEQ     .+4             ;BRANCH IF RESULT OK
3745  013620  104000                          HLT                     ;*ERROR* MFPS MODE 67 FAILED
3746                                                                  ;"TEMP" CONTAINS ERROR PROCESSOR STATUS
3747  013622  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3748
3750
3751
3752                                  ;*********************************************************************
3753                                  ;TEST SOB INSTRUCTION
3754                                  ;*********************************************************************
3755
3756  013624  012701  177776          SOB:    MOV     #-2,R1          ;LOOP COUNT COMPARE
3757  013630  012700  000002                  MOV     #2,R0           ;LOOP COUNT FOR SOB
3758  013634  005201                  1$:     INC     R1              ;UPDATE COMPARE COUNT
3759  013636  077002                          SOB     R0,1$           ;DONE LOOPING?
3760  013640  020001                          CMP     R0,R1           ;YES-TEST RESULT
3761  013642  001401                          BEQ     .+4             ;BRANCH IF OK
3762  013644  104000                          HLT                     ;*ERROR* SOB FAILED
3763  013646  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3764
3765                                  ;*********************************************************************
3766                                  ;TEST XOR INSTRUCTION MODE 0
3767                                  ;*********************************************************************
3768
3769  013650  012700  052525          XOR0:   MOV     #52525,R0       ;SET SOURCE = 52525
3770  013654  012701  125252                  MOV     #125252,R1      ;SET DEST = 125252
3771  013660  074001                          XOR     R0,R1           ;XOR 52525+125252
3772  013662  022701  177777                  CMP     #177777,R1      ;RESULT = 177777?
3773  013666  001401                          BEQ     .+4             ;BRANCH IF OK
3774  013670  104000                          HLT                     ;*ERROR* XOR MODE 0 FAILED
3775  013672  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3776
3777                                  ;*********************************************************************
3778                                  ;TEST XOR INSTRUCTION MODE NON 0 (MODE 67)
3779                                  ;*********************************************************************
3780
3781  013674  005000                  XOR67:  CLR     R0              ;SET SOURCE = 0
3782  013676  005067  001220                  CLR     TEMP            ;SET DEST = 0
3783  013702  074067  001214                  XOR     R0,TEMP         ;XOR 0+0
3784  013706  005767  001210                  TST     TEMP            ;IS RESULT = 0?
3785  013712  001401                          BEQ     .+4             ;BRANCH IF YES
3786  013714  104000                          HLT                     ;*ERROR* XOR MODE 67 FAILED
3787  013716  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3788
3789                                  ;*********************************************************************
3790                                  ;TEST SXT INSTRUCTION MODE 0 SIGN EXTEND 0'S
```

```
                                ;*****************************************************************
                                ;
013720  012700  177777  SXT00:  MOV     #-1,R0          ;SET DEST=177777
013724  000250          CLN             ;CLEAR N BIT
013726  006700          SXT     R0      ;SIGN EXTEND 0'S
013730  005700          TST     R0      ;IS RESULT = 0?
013732  001401          BEQ     .+4     ;BRANCH IF YES
013734  104000          HLT             ;*ERROR* SXT MODE 0 0'S FAILED
013736  104400          SCOPE           ;LOOP ON SUBTEST OR SETUP LOC. RETURN
                                ;
                                ;*****************************************************************
                                ;TEST SXT INSTRUCTION MODE NON 0 SIGN EXTEND 1'S
                                ;*****************************************************************
                                ;
013740  005067  001156  SXT61:  CLR     TEMP            ;SET DEST = 0
013744  000270          SEN             ;SET N BIT
013746  006767  001150          SXT     TEMP            ;SIGN EXTEND 1'S
013752  022767  177777  001142  CMP     #-1,TEMP        ;IS RESULT = -1?
013760  001401          BEQ     .+4     ;BRANCH IF YES
013762  104000          HLT             ;*ERROR* SXT MODE 67 1'S FAILED
013764  104400          SCOPE           ;LOOP ON SUBTEST OR SETUP LOC. RETURN
                                ;
                                ;*****************************************************************
                                ;TEST MARK INSTRUCTION
                                ;       TESTS SP INCREMENTS CORRECTLY
                                ;       TESTS PC LOADED CORRECT RETURN
                                ;       TESTS R5 LOADED CORRECTLY
                                ;*****************************************************************
                                ;
013766  010600  MARK0:  MOV     SP,R0           ;SAVE SP FOR COMPARE
013770  012705  000030          MOV     #1$-.-6,R5      ;SET "OLDR5" TO #1$ IN CURRENT BANK
013774  060705          ADD     PC,R5
013776  010537  000536          MOV     R5,@#OLDR5      ;SAVE OLDR5 FOR COMPARE
014002  010546          MOV     R5,-(SP)        ;PUSH "OLDR5" ONTO STACK
014004  012746  014042          MOV     #3$,-(SP)       ;PUSH 1 PARAMETER ONTO STACK
014010  012746  006401          MOV     #6401,-(SP)     ;PUSH "MARK 1" INST. ONTO STACK
014014  010605          MOV     SP,R5           ;POINT R5 TO MARK 1 INSTRUCTION
014016  012746  000004          MOV     #2$-.-6,-(SP)   ;SETUP GOOD RETURN ADDRESS IN CURRENT BANK
014022  060716          ADD     PC,(SP)
014024  000205          RTS     R5              ;DO THE MARK 1 INST. ON THE STACK AND CONT. AT 2$.
014026  104000  1$:     HLT                     ;*ERROR* MARK 1 FAILED
                                                ;RUN BASIC INSTRUCTION TEST TO SAVE YOU GRIEF WITH THIS
                                ;
014030  020600  2$:     CMP     SP,R0           ;WAS SP CORRECTLY RESTORED?
014032  001003          BNE     3$              ;BRANCH IF NO
014034  020537  000536          CMP     R5,@#OLDR5              ;WAS R5 CORRECTLY RESTORED (OLDR5)
014040  001404          BEQ     4$              ;BRANCH IF YES
014042  010637  015132  3$:     MOV     SP,@#MARKSP     ;SAVE FAILURE SP
014046  010006          MOV     R0,SP           ;RESTORE SP
014050  104000          HLT                     ;*ERROR* MARK 1 INST. FAILED
014052  104400  4$:     SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
                                ;
```

```
                                        ;*****************************************************************
                                        .SBTTL  EIS /FIS INSTRUCTION TESTS
                                        ;*****************************************************************
014054  032777  000010  001052          BIT     #BIT03,@SWR     ;EIS/FIS TEST WANTED?
014062  001402                          BEQ     ASH0            ;BRANCH IF YES
014064  000167  000522                  JMP     MATH            ;SKIP EIS/FIS TESTS

                                        ;*****************************************************************
                                        ;TEST ASH LEFT SHIFT 00001 BY 16 SETS C.
                                        ;*****************************************************************

014070  012700  000001          ASH0:   MOV     #1,R0           ;INIT DATA=00001
014074  000241                          CLC                     ;C BIT=0
014076  072027  000020                  ASH     #16.,R0         ;SHIFT 00001 LEFT 16.
014102  103001                          BCC     1$              ;BRANCH IF C BIT ERROR
014104  001401                          BEQ     .+4             ;BRANCH IF RESULT 0000
014106  104000                  1$:     HLT                     ;*ERROR* ASH LEFT SHIFT FAILED

                                        ;*****************************************************************
                                        ;TEST ASH RIGHT SHIFT #100000 BY 16. SETS C.
                                        ;*****************************************************************

014110  012700  100000          ASH1:   MOV     #100000,R0      ;INIT DATA=100000
014114  000241                          CLC                     ;C BIT=0
014116  072027  177760                  ASH     #-16.,R0        ;SHIFT 100000 RIGHT 16
014122  103003                          BCC     1$              ;BRANCH IF C BIT ERROR
014124  020027  177777                  CMP     R0,#177777      ;RESULT OK?
014130  001401                          BEQ     .+4             ;BRANCH IF YES
014132  104000                  1$:     HLT                     ;*ERROR* ASH RIGHT SHIFT FAILED
014134  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

                                        ;*****************************************************************
                                        ;TEST ASHC LEFT SHIFT 0000,0002 BY 31 SETS C.
                                        ;*****************************************************************

014136  005000          ASHC0:  CLR     R0              ;HIGH ORDER=0 (C BIT=0)
014140  012701  000002          MOV     #2,R1           ;LOW ORDER=2
014144  073027  000037          ASHC    #31.,R0         ;LEFT SHIFT 0002 BY 31
014150  103001                  BCC     1$              ;BRANCH IF C BIT ERROR
014152  001401                  BEQ     .+4             ;BRANCH IF RESULT=0
014154  104000          1$:     HLT                     ;*ERROR* ASHC LEFT SHIFT FAILED

                                        ;*****************************************************************
                                        ;TEST ASHC RIGHT SHIFT 100000,000000 BY 32 SETS C
                                        ;*****************************************************************

014156  012700  100000          ASHC1:  MOV     #100000,R0      ;HIGH ORDER=100000
014162  005001                  CLR     R1              ;LOW ORDER=0 (C BIT=0)
014164  073027  000040          ASHC    #32.,R0         ;RIGHT SHIFT 100000,0 BY 32
014170  103001                  BCC     1$              ;BRANCH IF C BIT ERROR
014172  100401                  BMI     .+4             ;BRANCH IF N BIT OK
014174  104000          1$:     HLT                     ;*ERROR* ASHC C OR N BIT FAILED
014176  022701  177777          CMP     #-1,R1          ;LOW ORDER RESULT OK?
014202  001003                  BNE     2$              ;BRANCH IF NO
014204  022700  177777          CMP     #-1,R0          ;HIGH ORDER RESULT OK?
014210  001401                  BEQ     .+4             ;BRANCH IF YES
```

```
3929  014212  104000                2$:     HLT                     ;*ERROR* ASHC RIGHT SHIFT FAILED
3930  014214  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3931
3932                                ;***********************************************************************
3933                                ;TEST MUL INSTRUCTION ALT 1'S
3934                                ;***********************************************************************
3935
3936                                ;       125252*40000=165252,100000
3937
3938  014216  012700  125252        MUL0:   MOV     #125252,R0      ;INIT MULTIPLICAND
3939  014222  070027  040000                MUL     #40000,R0       ;R1:R0=40000*125252
3940  014226  100003                        BPL     1$              ;BRANCH IF N BIT ERROR
3941  014230  102402                        BVS     1$              ;BRANCH IF V BIT ERROR
3942  014232  001401                        BEQ     1$              ;BRANCH IF Z BIT ERROR
3943  014234  103401                        BCS     .+4             ;BRANCH IF C BIT OK
3944  014236  104000                1$:     HLT                     ;*ERROR* MUL FLAGS FAILED
3945  014240  022700  165252                CMP     #165252,R0      ;HIGH ORDER RESULT OK?
3946  014244  001003                        BNE     2$              ;BRANCH IF NO
3947  014246  022701  100000                CMP     #100000,R1      ;LOW ORDER RESULT OK?
3948  014252  001401                        BEQ     .+4             ;BRANCH IF YES
3949  014254  104000                2$:     HLT                     ;*ERROR* MUL FAILED
3950  014256  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3951
3952                                ;***********************************************************************
3953                                ;TEST DIV INSTRUCTION ALT 1'S
3954                                ;***********************************************************************
3955
3956  014260  012701  125252        DIV0:   MOV     #125252,R1      ;LOW ORDER DIVIDEND
3957  014264  012700  177777                MOV     #-1,R0          ;HIGH ORDER DIVIDEND
3958  014270  000261                        SEC                     ;C BIT=1
3959  014272  071027  000002                DIV     #2,R0           ;R1:R0=177777,125252/2
3960  014276  103403                        BCS     1$              ;BRANCH IF C BIT ERROR
3961  014300  102402                        BVS     1$              ;BRANCH IF V BIT ERROR
3962  014302  001401                        BEQ     1$              ;BRANCH IF Z BIT ERROR
3963  014304  100401                        BMI     .+4             ;BRANCH IF N BIT OK
3964  014306  104000                1$:     HLT                     ;*ERROR* DIV FLAGS FAILED
3965  014310  005701                        TST     R1              ;LOW ORDER RESULT OK?
3966  014312  001003                        BNE     2$              ;BRANCH IF NO
3967  014314  020027  152525                CMP     R0,#152525      ;LOW ORDER RESULT OK?
3968  014320  001401                        BEQ     .+4             ;BRANCH IF YES
3969  014322  104000                2$:     HLT                     ;*ERROR* DIV FAILED
3970  014324  104400                        SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
3971
```

```
                                ;*****************************************************************
                                ;FIS BASIC INSTRUCTION TESTS
                                ;FADD, FSUB, FMUL, FDIV


                                ;TEST FADD INSTRUCTION

                                ;THE TEST ADDS TWO NUMBERS THAT HAVE EXPONENTS 27 APART,
                                ;WHICH IS THE LARGEST DIFFERENCE ALLOWED.

                                ;       71600,000000 + 77577,177777= 77600,000000
                                ;OR IN ENGLISH (INCLUDES HIDDEN NORMALIZE BIT)
                                ;       .1 E350 + .57777777 E377=.1 E351
                                ;
                                ;CC=0000
                                ;NOTE; THE FIS FLOATING STACK IS ALWAYS IN BANK 0
                                ;*****************************************************************

014326  012701  014612          FADDO:  MOV     #FSTACK+10,R1   ;INIT FLOATING STACK
014332  005041                          CLR     -(R1)           ;INIT LOW ARGUMENT A
014334  012741  071600                  MOV     #71600,-(R1)    ;INIT HIGH ARGUMENT A
014340  012741  177777                  MOV     #-1,-(R1)       ;INIT LOW ARGUMENT B
014344  012741  077577                  MOV     #77577,-(R1)    ;INIT HIGH ARGUMENT B
014350  000262                          SEV                     ;V=1
014352  000261                          SEC                     ;C=1
014354  075001                          FADD    R1              ;ADD NO.S IN HEADING
014356  100403                          BMI     1$              ;N BIT=0
014360  001402                          BEQ     1$              ;Z BIT=0
014362  102401                          BVS     1$              ;V BIT=0
014364  103001                          BCC     .+4             ;C BIT=0
014366  104000                  1$:     HLT                     ;*ERROR* FADD STATUS FAILED
014370  022737  077600  014606          CMP     #77600,@#HIGHA  ;HIGH ANSWER OK?
014376  001003                          BNE     2$              ;BRANCH IF NO
014400  005737  014610                  TST     @#LOWA          ;LOW ORDER ANSWER OK?
014404  001401                          BEQ     .+4             ;BRANCH IF YES
014406  104000                  2$:     HLT                     ;*ERROR* FADD FAILED
014410  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN

                                ;*****************************************************************
                                ;TEST FSUB INSTRUCTION

                                ;       FSUB MICROCODE IS SAME AS FADD EXCEPT FOR 1 LOCATION.
                                ;THIS TEST WILL TEST MORE FADD MICROCODE BY MAKING ARGUMENT A 1
                                ;GREATER THAN ARGUMENT B.

                                ;       125252,125253-125252,125252=117400,000000
                                ;CC=1000 (NOT TESTED)
                                ;NOTE; THE FIS FLOATING STACK IS ALWAYS IN BANK 0
                                ;*****************************************************************

014412  012701  014612          FSUBO:  MOV     #FSTACK+10,R1   ;INIT FLOATING STACK
014416  012700  125253                  MOV     #125253,R0
014422  010041                          MOV     R0,-(R1)        ;LOW ARGUMENT A=125253
014424  005300                          DEC     R0
014426  010041                          MOV     R0,-(R1)        ;HIGH ARGUMENT A=125252
```

```
4029  014430  010041                          MOV     R0,-(R1)        ;LOW ARGUMENT B=125252
4030  014432  010041                          MOV     R0,-(R1)        ;HIGH ARGUMENT B=125252
4031  014434  075011                          FSUB    R1              ;SUBTRACT NO.S IN HEADING
4032  014436  022737  117400  014606          CMP     #117400,@#HIGHA ;HIGH ANSWER OK?
4033  014444  001003                          BNE     1$              ;BRANCH IF NO
4034  014446  005737  014610                  TST     @#LOWA          ;LOW ANSWER OK
4035  014452  001401                          BEQ     .+4             ;BRANCH IF YES
4036  014454  104000                  1$:     HLT                     ;*ERROR* FSUB FAILED
4037  014456  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
4038
4039                                  ;*********************************************************************
4040                                  ;TEST FMUL INSTRUCTION
4041
4045                                  ;TESTS SLOW MULTIPLY MICROCODE
4046
4047                                  ;       161616,161616*000052,125252=000000,000000
4048                                  ;CC=0100 (NOT TESTED)
4049                                  ;NOTE; THE FIS FLOATING STACK IS ALWAYS IN BANK 0
4050                                  ;*********************************************************************
4051
4052  014460  012701  014612          FMULO:  MOV     #FSTACK+10,R1   ;INIT FLOATING STACK
4053  014464  012741  161616                  MOV     #161616,-(R1)   ;LOW ARGUMENT A=161616
4054  014470  012741  161616                  MOV     #161616,-(R1)   ;HIGH ARGUMENT A=161616
4055  014474  012741  125252                  MOV     #125252,-(R1)   ;LOW ARGUMENT B=125252
4056  014500  012741  000052                  MOV     #000052,-(R1)   ;HIGH ARGUMENT B=000052
4057  014504  075021                          FMUL    R1              ;FMUL NO.S IN HEADING
4058  014506  005737  014606                  TST     @#HIGHA         ;HIGH ANSWER OK?
4059  014512  001003                          BNE     1$              ;BRANCH IF NO
4060  014514  005737  014610                  TST     @#LOWA          ;LOW ANSWER OK?
4061  014520  001401                          BEQ     .+4             ;BRANCH IF YES
4062  014522  104000                  1$:     HLT                     ;*ERROR* FMUL FAILED
4063  014524  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
4064
4065                                  ;*********************************************************************
4066                                  ;TEST FDIV INSTRUCTION
4067
4071
4072                                  ;       167452,125252/027652,125253=177777,177777
4073                                  ;CC=1000 (NOT TESTED)
4074                                  ;NOTE; THE FIS FLOATING STACK IS ALWAYS IN BANK 0
4075                                  ;*********************************************************************
4076
4077  014526  012701  014612          FDIVO:  MOV     #FSTACK+10,R1   ;INIT FLOATING STACK
4078  014532  012741  125252                  MOV     #125252,-(R1)   ;LOW ARGUMENT A=125252
4079  014536  012741  167452                  MOV     #167452,-(R1)   ;HIGH ARGUMENT A=167452
4080  014542  012741  125253                  MOV     #125253,-(R1)   ;LOW ARGUMENT B=125253
4081  014546  012741  027652                  MOV     #027652,-(R1)   ;HIGH ARGUMENT B=027652
4082  014552  075031                          FDIV    R1              ;DIVIDES NO.S IN HEADING
4083  014554  022737  177777  014606          CMP     #-1,@#HIGHA     ;HIGH ANSWER OK?
4084  014562  001004                          BNE     1$              ;BRANCH IF NO
4085  014564  022737  177777  014610          CMP     #-1,@#LOWA      ;LOW ANSWER OK?
4086  014572  001401                          BEQ     .+4             ;BRANCH IF YES
4087  014574  104000                  1$:     HLT                     ;*ERROR* FDIV FAILED
4088  014576  104400                          SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
4089  014600  000404                          BR      MATH            ;SKIP OVER FLOATING STACK
```

```
                                ;*FLOATING ARGUMENT STACK
                                ;*THIS STACK IS ALWAYS IN MEMORY BANK 0

014602  000000                  FSTACK: .WORD   0               ;HIGH ARGUMENT B
014604  000000                          .WORD   0               ;LOW ARGUMENT B
014606  000000                  HIGHA:  .WORD   0               ;HIGH ARGUMENT A + RESULT
014610  000000                  LOWA:   .WORD   0               ;LOW ARGUMENT A + RESULT




                                ;*********************************************************************
                                ;MATH
                                ;       A TEST OF THE COMPLEMENTING ARITHMETIC INSTRUCTIONS.
                                ;E.G. INC-DEC, NEG-COM-INC, ADC-SBC, ADD-SUB, ASR-ASL
                                ;THE TEST WILL LOOP 10000 OCTAL  TIMES REGARDLESS IF
                                ;ITERATION IS ENABLED OR DISABLED.

                                ;
                                ;REGISTER USAGE
                                ;       R0= CONTAINS THE BASE NO. USED IN THE TESTING.
                                ;       R1= WORKING REGISTER FOR MOST ARITHMETIC INSTRUCTIONS TESTED.
                                ;       R2= NOT USED (RESERVED FOR FLOPPY TEST)
                                ;       R3= USED IN ASL/ASR TEST
                                ;       R4= 1
                                ;       R5= 10000       ;PASS COUNT
                                ;*********************************************************************

014612  012737  000001  016212  MATH:   MOV     #1,@#ICOUNT     ;DISABLE  SR ITERATIONS
014620  012700  000177          MOV     #177,R0         ;INIT BASE DATA
014624  012704  000001          MOV     #1,R4           ;SET R4=00001
014630  012705  010000          MOV     #10000,R5       ;SETUP NO. OF ITERATIONS.

014634  162700  000177          MATHLP: SUB     #177,R0         ;R0=R0-177 (FIRST TIME=0)

014640  010001                  MOV     R0,R1
014642  005201                  1$:     INC     R1              ;R1=R1+1
014644  005201                          INC     R1
014646  005201                          INC     R1
014650  005201                          INC     R1
014652  005201                          INC     R1
014654  005201                          INC     R1
014656  005201                          INC     R1
014660  005201                          INC     R1              ;R1=R1+10
014662  005301                          DEC     R1              ;R1=R1-1
014664  005301                          DEC     R1
014666  005301                          DEC     R1
014670  005301                          DEC     R1
014672  005301                          DEC     R1
014674  005301                          DEC     R1
014676  005301                          DEC     R1
014700  005301                          DEC     R1              ;R1=R1-10
```

```
 4147  014702  020100                    CMP     R1,R0           ;R1 RESTORED OK?
 4148  014704  001401                    BEQ     .+4             ;BRANCH IF YES
 4149  014706  104000                    HLT                     ;*ERROR* INC OR DEC FAILED
 4150
 4151  014710  005401                    NEG     R1
 4152  014712  005101                    COM     R1              ;SIMULATE NEG R1
 4153  014714  005201                    INC     R1              ;SIMULATE NEG R1
 4154  014716  020100                    CMP     R1,R0           ;R1 RESTORED FROM NEGATE?
 4155  014720  001401                    BEQ     .+4             ;BRANCH IF YES
 4156  014722  104000                    HLT                     ;*ERROR* NEG OR COM FAILED
 4157  014724  000261                    SEC                     ;SET C BIT
 4158  014726  005501                    ADC     R1              ;ADD C BIT TO R1
 4159  014730  000261                    SEC
 4160  014732  005601                    SBC     R1              ;SUBTRACT C BIT FROM R1
 4161  014734  020100                    CMP     R1,R0           ;R1 RESTORED CORRECTLY?
 4162  014736  001401                    BEQ     .+4             ;BRANCH IF YES
 4163  014740  104000                    HLT                     ;*ERROR* ADC OR SBC FAILED
 4164
 4165  014742  060001                    ADD     R0,R1           ;R1=R1+R0
 4166  014744  160001                    SUB     R0,R1           ;R1=R1-R0
 4167  014746  020100                    CMP     R1,R0           ;R1 RESTORED CORRECTLY?
 4168  014750  001401                    BEQ     .+4             ;BRANCH IF YES
 4169  014752  104000                    HLT                     ;*ERROR* ADD OR SUB FAILED
 4170
 4171  014754  010103                    MOV     R1,R3           ;TEST ASR/ASL
 4172  014756  006203                    ASR     R3              ;SHIFT R3 RIGHT 1
 4173  014760  006303                    ASL     R3              ;SHIFT R3 LEFT 1
 4174  014762  040401                    BIC     R4,R1           ;CLEAR BIT 00
 4175  014764  020103                    CMP     R1,R3           ;R3 RESTORED CORRECTLY?
 4176  014766  001401                    BEQ     .+4             ;BRANCH IF YES
 4177  014770  104000                    HLT                     ;*ERROR* ASR OR ASL FAILED
 4178  014772  077560                    SOB     R5,MATHLP       ;REPEAT TEST
 4179
 4180  014774  104400          MATHND: SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
 4181
 4275
 4276
 4277                          ;******************************************************************
 4278                          ;IF POWER FAIL OCCURRED - RESTART TEST FROM HERE
 4279                          ;******************************************************************
 4280
 4281  014776  005737  016330          TST     @#PWRFLG        ;SET ON POWER FAIL
 4282  015002  001404                  BEQ     WAIT5           ;SKIP OVER IF CLEAR
 4283  015004  005037  016330          CLR     @#PWRFLG
 4284  015010  000137  000666          JMP     @#ESTART        ;POWER FAIL OCCURRED SO RESTART
 4285
 4286                          .SBTTL IF T BIT PASS SKIP INTERRUPT OUT OF WAIT TEST
 4287
 4288  015014  005737  015532  WAIT5:  TST     @#TRPB          ;IS THIS A T BIT PASS?
 4289  015020  100422                  BMI     WAITX           ;BRANCH IF YES
 4290
 4291
 4292  015022  012737  000010  016212          MOV     #10,@#ICOUNT
 4293
 4294                          ;******************************************************************
 4295                          ;TEST INTERRUPTS OUT OF WAIT INSTRUCTION
```

```
4296                                    ;*****************************************************************
4297
4298  015030  017701  000100            WAITST: MOV     @SWR,R1         ;GET SWITCH SETTINGS.
4299  015034  042701  000230                    BIC     #CWAIT,R1       ;CLEAR UNUSED BITS AND PLU/EIS BITS
4300  015040  122701  177547                    CMPB    #↑CCWAIT,R1     ;SELECTED DEVICES STORED IN SWREG
4301  015044  001404                            BEQ     WAIT4           ;BRANCH IF NO DEVICES SELECTED
4302  015046  000001                            WAIT                    ;INTERRUPTS WILL OCCUR
4303  015050  000001                            WAIT                    ;IF DEVICES ARE SELECTED
4304  015052  000001                            WAIT
4305  015054  000001                            WAIT
4306  015056  104400                    WAIT4:  SCOPE                   ;LOOP ON SUBTEST OR SETUP LOC. RETURN
4307  015060  012737  001000  016212            MOV     #1000,@#ICOUNT  ;RESET ITERATION COUNTER
4308  015066  000442                    WAITX:  BR      EAESRT          ;SKIP OVER STORAGE AREA
4309
4310
4311                                    ;*****************************************************************
4312                                    .SBTTL BACKGROUND INSTRUCTION TEST TEMP STORAGE AND CONSTANTS
4313                                    ;*****************************************************************
4314
4315
4375
4376  015070  125252                    B:      125252
4377  015072  015070                            B                       ;ADDRESS OF B
4378  015074  052525                            052525
4379
4380          015100                    .=B+10
4381  015100  177777                    A:      -1
4382  015102  015104                            A+4
4383
4384          015104                    .=A+4
4385  015104  125252                            125252
4386  015106  015110                            A+10                    ;ADDRESS OF A+10
4387  015110  052525                            052525
4388                                    ;FOR STORAGE
4389  015112  000000                    C:      0
4390  015114  015112                            C                       ;ADDRESS OF C
4391
4392          015122                    .=C+10
4393  015122  000000                    TEMP:   0
4394  015124  015122                            TEMP                    ;ADDRESS OF TEMP
4395
4396          015130                    .=TEMP+6
4397  015130  015132                            TEMP+10                 ;ADDRESS OF TEMP+10 OR "D"
4398  015132                            MARKSP:
4399  015132  000000                    D:      0
4400
4401                                    ;SWR MUST BE HERE SO IT CAN BE RELOCATED
4402
4403  015134  177570                    SWR:    177570                  ;CHANGES TO 176 IF NO HARDWARE SR PRESENT
4404
4405
4406
4407                                    ;GROUP OF NESTED SUBROUTINES FOR JSR TESTING
4408
4409  015136  000207                    SUBR1:  RTS     PC              ; ONE INSTRUCTION
4410  015140  000277                    SUBR2:  SCC
```

```
4411  015142  000205                          RTS     R5                  ;ONE SUBROUTINE LEVELS
4412  015144  004537  015140        SUBR3:    JSR     R5,@#SUBR2          ;TWO SUBROUTINE LEVELS
4413  015150  000204                          RTS     R4
4414  015152  004467  177766        SUBR4:    JSR     R4,SUBR3            ;THREE SUBROUTINE LEVELS
4415  015156  000203                          RTS     R3
4416  015160  004367  177766        SUBR5:    JSR     R3,SUBR4            ;FOUR SUBROUTINE LEVELS
4417  015164  000200                          RTS     R0
4418  015166  004067  177766        SUBR6:    JSR     R0,SUBR5            ;FIVE SUBROUTINE LEVELS
4419  015172  000207                          RTS     PC                  ;AND EXIT
4420
4421
4422  015174  012737  016232  000024 EAESRT:  MOV     #PFAIL,@#24         ;SETUP PWR FAIL VECTOR
4423  015202  012737  000340  000026          MOV     #340,@#26
4424
4425                                ;IF RELOCATION ENABLED THEN TRPA IS MODIFIED IN THE UPPER BANKS.
4426                                ;       FOR EXAMPLE IF 12K UNDER TEST THEN TRPA+20000=JMP @#BEGIN+40000
4427                                ;       AND TRPA+4000=JMP @#BANK0
4428  015210  000137  015214        TRPA:   JMP     @#BANK0             ;SEE NOTE ABOVE
4429  015214                        ENDREL:                             ;LAST RELOCATED LOCATION
4430
4431
4432                                ;******************************************************************
4433                                .SBTTL ***END RELOCATED INSTRUCTION TESTS ***
4434                                .ENABL AMA
4435                                .SBTTL
4436                                ;******************************************************************
4437
4438
4439
4440
4441                                ;******************************************************************
4442                                .SBTTL INCREMENT PASS COUNT AND PRINT END OF PASS DATA
4443                                ;******************************************************************
4444
4445
4446  015214  005237  000574        BANK0:  INC     PASSN1              ;INCR INST PASS COUNTER
4447  015220  023737  000576  000574        CMP     PASSN0,PASSN1       ;TIME TO INCR PRINTED PASS COUNT?
4448  015226  103063                        BHIS    YESTR               ;BRANCH IF NO
4449
4450                                ;CHECK THAT RX AND PLU MADE A PASS
4451
4452  015230  005037  000532                CLR     RETURN              ;IN CASE OF ERROR HALT
4453  015234  017700  177674                MOV     @SWR,R0             ;GET DEVICES INHIBITED
4454  015240  005100                        COM     R0                  ;MAKE ENABLES FOR CHECK
4455  015242  042700  177754                BIC     #^CCHECKM,R0        ;CLEAR UNWANTED BITS
4456  015246  043700  000530                BIC     CHECK,R0            ;CLEAR BITS OF DEVICES THAT PASSED
4457  015252  001401                        BEQ     .+4                 ;BRANCH IF DEVICES SELECTED PASSED.
4458  015254  104000                        HLT                         ;*ERROR* PLU SELECTED BUT DID NOT MAKE A PASS
4459                                                                    ;POSSIBLE CAUSES:
4460                                                                    ;FLOPPY FAILED TO INTERRUPT
4461                                                                    ;PLU TEST CABLE NOT INSTALLED.
4462  015256  005237  000524                INC     PASS                ;INCREMENT TOTAL PASS COUNTER.
4463  015262  005037  000574                CLR     PASSN1              ;INIT RUNNING PASS COUNT
4464                                ;******************************************************************
4465                                ;PRINT PASS=XXXX ERROR=XXXX RXERROR=XXXX TIME=XXXX
4466                                ;******************************************************************
```

```
4467
4468  015266  105777  177642          ENDPAS: TSTB    @SWR            ;END OF PASS PRINTOUT WANTED?
4469  015272  100441                          BMI     YESTR           ;BRANCH IF NO
4471  015274  106700                          MFPS    R0              ;SAVE CURRENT STATUS
4472  015276  106427  000340                  MTPS    #340            ;LOCK OUT INT.
4474
4479  015302  104001  017162                  TYPE    ,PAS            ;TYPE "PASS="
4480  015306  013703  000524                  MOV     PASS,R3         ;GET PASS COUNT
4481  015312  004737  016012                  JSR     PC,TYPOCT       ;PRINT PASS COUNT IN OCTAL
4482  015316  104001  017171                  TYPE    ,ERR            ;TYPE "ERROR="
4483  015322  013703  000526                  MOV     ERROR,R3        ;GET ERROR COUNT
4484  015326  004737  016012                  JSR     PC,TYPOCT       ;TYEP ERROR COUNT
4485  015332  104001  017150                  TYPE    ,RXS            ;TYPE "RXERROR="
4486  015336  013703  000516                  MOV     RXSOFT,R3       ;GET RX SOFT ERROR COUNT
4487  015342  004737  016012                  JSR     PC,TYPOCT       ;TYPE RXSOFT ERROR COUNT
4488  015346  104001  017201                  TYPE    ,TIM            ;TYPE "TIME="
4489  015352  013703  003576                  MOV     TIME,R3         ;GET CLOCK TIME
4490  015356  004737  016012                  JSR     PC,TYPOCT       ;TYPE TIME IN OCTAL
4491  015362  104001  017146                  TYPE    ,CR             ;TYPE CRLF
4492
4494  015366  106400                          MTPS    R0              ;RESTORE PROC. STATUS
4496
4500  015370  005037  002130                  CLR     TPBDAT          ;INIT CONSOLE DATA FOR CLEAN RESTART
4501  015374  000414                          BR      CLRTBT          ;CLR TBIT IF SET AND RESTART
4502
4503                                  ;*****************************************************************
4504                                  .SBTTL  TEST FOR T BIT TRAP SETUP
4505                                  ;*****************************************************************
4506
4507  015376  005737  015532          YESTR:  TST     TRPB            ;PREVIOUS PASS WITH T BIT?
4508                                                                  ;TRPB IS ALSO END OF RELOCATION TEST PASS FLAG.
4509  015402  001411                          BEQ     CLRTBT          ;BRANCH IF NO
4510
4511  015404  032777  001000  177522  1$:     BIT     #BIT09,@SWR     ;RELOCATION ENABLED?
4512  015412  001005                          BNE     CLRTBT          ;BRANCH IF NO
4513  015414  005737  000540                  TST     @#CHAIN         ;RXDP CHAIN MODE?
4514  015420  001002                          BNE     CLRTBT          ;BRANCH IF YES (NO RELOCATION TEST)
4515  015422  000137  023600                  JMP     @#BEGIN+20000   ;START INSTRUCTION TEST IN BANK1
4516
4517  015426  005016                  CLRTBT: CLR     (SP)            ;SETUP STACK FOR POSSIBLE TBIT PASS
4518  015430  005137  015532                  COM     TRPB            ;SET T BIT SWITCH EVERY OTHER PASS
4519  015434  032777  010000  177472          BIT     #BIT12,@SWR     ;T BIT TRAP TEST WANTED
4520  015442  001015                          BNE     LINKER          ;BRANCH IF NO
4521  015444  012737  015534  000014          MOV     #YESRT,@#14     ;SETUP T BIT TRAP VECTOR
4522  015452  005037  000016                  CLR     @#16            ;CLR TBIT BUT ALLOW INTERRUPTS
4523  015456  005737  015532                  TST     TRPB            ;T BIT TEST NEXT PASS?
4524  015462  100005                          BPL     LINKER          ;BRANCH IF NO
4525  015464  012716  000020                  MOV     #20,(SP)        ;SET T BIT ON STACK
4526  015470  012746  003600          YESTR1: MOV     #BEGIN,-(6)     ;RESTART TEST
4527  015474  000002                          RTI                     ;GOTO BEGIN VIA RTI
4528
4529                                  ;*****************************************************************
4530                                  .SBTTL ACT11 END OF PASS ROUTINE
4531                                  ;*****************************************************************
4532
4533  015476  105737  000042          LINKER: TSTB    @#42            ;SHOULD PROGRAM GO TO MONITOR
```

```
4534  015502  001772                        BEQ     YESTR1          ;BRANCH IF NO ACT11 (REPEAT PROCESSOR TEST)
4535  015504  012737  000016  000014        MOV     #16,@#14        ;LOAD T BIT TRAP VECTOR
4536  015512  013700  000042        ACT:    MOV     @#42,R0         ;GO TO MONITOR OR ACTII SYSTEM
4537  015516  004710                $ENDAD: JSR     PC,(R0)         ;RETURN TO ACT11
4538  015520  000240                        NOP                     ;IF QUICK VERIFY=RESET ELSE NOP
4539  015522  000240                        NOP                     ;IF QUICK VERIFY=CLR #-1 ELSE INC #0
4540  015524  000240                        NOP                     ;IF QUICK VERIFY=BR .-4 ELSE NOP
4541  015526  000137  000666              JMP     @#ESTART        ;REPEAT TEST UNDER ACT11
4542  015532  000000        TRPB:   0
4543  015534  000006        YESRT:  RTT                             ;RETURN TO PROGRAM FROM T BIT TRAP
4544  015536  000000                        HALT                    ;RTI FAILED
4545
4546
4547                          ;******************************************************************
4548                          .SBTTL ***MESSAGE/ERROR SERVICE SUBROUTINES***
4549
4550                          ;HLTMES- ENTERED IF HLT OR TYPE TRAP
4551                          ;       DECODES HLT OR TYPE AND GOES TO APPRO. ROUTINE
4552                          ;******************************************************************
4553
4554  015540  010137  000552        HLTMES: MOV     R1,SAVR1        ;SAVE R1
4555  015544  010037  000550                MOV     R0,SAVR0        ;SAVE R0
4556  015550  011601                        MOV     (SP),R1         ;POINT R1 TO TRAP CALL
4557  015552  005741                        TST     -(R1)           ;POINT R1 TO TRAP PC
4558  015554  011100                        MOV     (R1),R0         ;GET TRAP CALL
4559  015556  006000                        ROR     R0              ;IF BIT 0 THEN TYPE CALL.
4560  015560  103467                        BCS     MES             ;BRANCH IF TYPE TRAP CALL
4561  015562  013700  000550                MOV     SAVR0,R0        ;RESTORE R0
4562  015566  013701  000552                MOV     SAVR1,R1        ;RESTORE R1
4563                                                                ;FALL INTO PRINT ROUTINE
4564                          ;******************************************************************
4565                          ;PRINT PC=XXXX PSW=XXXX LOOP=XXXX
4566                          ;******************************************************************
4567
4568  015572  005237  000526        PERROR: INC     ERROR           ;ERROR COUNT LOCATION
4569  015576  037727  177332  020000        BIT     @SWR,#20000     ;INHIBIT ERROR PRINTOUT?
4570  015604  001036                        BNE     1$              ;BRANCH IF NO
4571  015606  010637  000564                MOV     SP,SAVSP        ;SAVE SP FOR POSTERITY
4572  015612  062737  000004  000564        ADD     #4,@#SAVSP      ;MAKE SP WHAT IT WAS AT TIME OF ERROR
4573  015620  012637  000566                MOV     (6)+,SAVPC      ;PC OF FAILING ROUTINE
4574  015624  012637  000570                MOV     (6)+,SAVPS      ;CC OF ERROR CONDITION
4575  015630  024646                        CMP     -(6),-(6)       ;REPOSITION THE STACK
4576  015632  104001  017111                TYPE    ,PCE            ;TYPE "PC="
4577  015636  013703  000566                MOV     $AVPC,R3
4578  015642  004737  016012                JSR     PC,TYPOCT       ;PRINT OCTAL NUMBER
4579  015646  104001  017116                TYPE    ,CCE            ;TYPE "CC="
4580  015652  013703  000570                MOV     $AVPS,R3
4581  015656  004737  016012                JSR     PC,TYPOCT       ;PRINT OCTAL NUMBER
4582  015662  104001  017135                TYPE    ,SCPE           ;TYPE "LOOP="
4583  015666  013703  000532                MOV     RETURN,R3       ;SCOPE LOOP BEGIN ADDRESS
4584  015672  004737  016012                JSR     PC,TYPOCT
4585  015676  104001  017146                TYPE    ,CR             ;CRLF
4586  015702  005777  177226        1$:     TST     @SWR            ;USER WANT HALT ON ERROR?
4587  015706  100404                        BMI     2$              ;BRANCH IF YES
4588  015710  023727  000042  015516        CMP     @#42,#$ENDAD    ;ACT 11 AUTO MODE?
4589  015716  001001                        BNE     .+4             ;BRANCH IF NO (DON'T HALT ON ERROR)
```

```
4590  015720  000000                    2$:     HWLT                    ;*NORMAL HALT* HALT ON ERROR SWITCH SET
4591
4592  015722  032777  000400  177204            BIT     #400,@SWR       ;USER WANT RESTART ON ERROR?
4593  015730  001402                            BEQ     3$              ;BRANCH IF NO
4594  015732  000137  000666                    JMP     ESTART          ;RESTART ON ERROR
4595  015736  000002                    3$:      RTI                     ;RETURN TO MAIN LINE PROCESSOR TEST
4596
4597
4598                                    ;*********************************************************************
4599                                    ;TYPE- TRAP CALL SERVICE ROUTINE
4600                                    ;       COME HERE FROM HLTMES ROUTINE.
4601                                    ;REGISTERS USED:
4602                                    ;       R0- HOLDS CHAR
4603                                    ;       R1- POINTS TO BUFFER
4604                                    ;NOTE; R0 AND R1 ARE RESTORED HERE AND WERE SAVED IN HLTMES
4605                                    ;*********************************************************************
4606
4607
4608  015740  005721                    MES:    TST     (R1)+           ;POINT TO MESSAGE BUFFER ADDRESS
4609  015742  011101                            MOV     (R1),R1         ;POINT R1 TO MESSAGE BUFFER
4610  015744  112100                    1$:     MOVB    (R1)+,R0        ;GET A CHAR FROM BUFFER.
4611  015746  001412                            BEQ     3$              ;BRANCH IF END OF BUFFER
4612  015750  120027  000045                    CMPB    R0,#'%          ;IS IT A %?
4613  015754  001005                            BNE     2$              ;NO
4614  015756  012700  000015                    MOV     #15,R0          ;PRINT CR
4615  015762  000004                            PNTCHR
4616  015764  012700  000012                    MOV     #12,R0          ;PRINT LF
4617  015770  000004                    2$:     PNTCHR
4618  015772  000764                            BR      1$              ;GET ANOTHER CHAR.
4619
4620  015774  013700  000550            3$:     MOV     SAVR0,R0        ;RESTORE R0 SAVED IN HLTMES
4621  016000  013701  000552                    MOV     SAVR1,R1        ;RESTORE R1 SAVED IN HLTMES
4622  016004  062716  000002                    ADD     #2,(SP)         ;BUMP SP PAST MESSAGE ADDRESS
4623  016010  000002                            RTI
4624
4625
4626
4627
4628                                    ;*********************************************************************
4629                                    .SBTTL OCTAL TYPEOUT ROUTINE
4630                                    ;REGISTER USAGE;
4631                                    ;R0-R5 SAVED IN CORE
4632                                    ;R0= CHAR PRINTED
4633                                    ;R1= CHAR COUNT (ALWAYS 6)
4634                                    ;R3= OCTAL NO. TO BE PRINTED ON ENTRY
4635                                    ;*********************************************************************
4636  016012  010037  000550            TYPOCT: MOV     R0,SAVR0        ;SAVE R0
4637  016016  010137  000552                    MOV     R1,SAVR1        ;SAVE R1
4638  016022  010237  000554                    MOV     R2,SAVR2        ;SAVE R2
4639  016026  010337  000556                    MOV     R3,SAVR3        ;SAVE R3
4640  016032  010437  000560                    MOV     R4,SAVR4        ;SAVE R4
4641  016036  112701  000006                    MOVB    #6,R1           ;CHAR. COUNT
4642  016042  005000                            CLR     R0              ;HOLDS CHAR
4643  016044  000241                    4$:     CLC
4644  016046  006103                            ROL     R3              ;GET BIT 15 INTO C BIT
4645  016050  006100                            ROL     R0              ;PUT IT INTO R0
```

```
 4646  016052  052700  000060                BIS     #60,R0          ;MAKE IT ASCII
 4647  016056  000004                        PNTCHR                  ;AND PRINT IT
 4648  016060  005000                        CLR     R0              ;INIT CHAR HOLDER
 4649  016062  006103                        ROL     R3              ;PUT EACH BIT OF THE NEXT OCTAL CHAR INTO R0.
 4650  016064  006100                        ROL     R0
 4651  016066  006103                        ROL     R3
 4652  016070  006100                        ROL     R0
 4653  016072  105301                        DECB    R1              ;DONE ALL CHARS?
 4654  016074  001363                        BNE     4$              ;BRANCH IF NO
 4655
 4656
 4657  016076  013700  000550                MOV     SAVR0,R0
 4658  016102  013701  000552                MOV     SAVR1,R1
 4659  016106  013703  000556                MOV     SAVR3,R3
 4660  016112  000207                        RTS     PC              ;RETURN TO CALLER
 4661
 4662                          ;******************************************************************
 4663                          .SBTTL SCOPE LOOP ROUTINE
 4664                          ;SCOPE LOOP ROUTINE ENTERED BY USER TRAP ".SCOPE"
 4665                          ;IF PLU SELECTED SET PLU INTERRUPT ENABLE
 4666                          ;******************************************************************
 4667
 4668
 4669  016114  032777  040000  177012  SCOPEC: BIT     #BIT14,@SWR     ;TEST SR FOR SCOPE
 4670  016122  001012                        BNE     2$              ;BRANCH IF SCOPE LOOP  WANTED.
 4671  016124  032777  004000  177002        BIT     #BIT11,@SWR     ;ITERATION WANTED?
 4672  016132  001011                        BNE     3$              ;BRANCH IF NO
 4673  016134  023737  016214  016212  1$:   CMP     SCOPEF,ICOUNT   ;DONE ITERATIONS?
 4674  016142  001405                        BEQ     3$              ;BRANCH IF YES
 4675  016144  005237  016214                INC     SCOPEF          ;UPDATE ITERATION COUNT.
 4676  016150  013716  000532          2$:   MOV     RETURN,@SP      ;REPOSITION THE STACK
 4677  016154  000404                        BR      4$              ;SCOPE RETURN
 4678
 4679
 4680  016156  005037  016214          3$:   CLR     SCOPEF          ;CLR ITERATION COUNTER
 4681  016162  011637  000532                MOV     @SP,RETURN      ;SAVE BEGIN OF NEXT TEST ADDRESS
 4682
 4683                          ;IF PLU TESTED SET INTERRUPT ENABLE
 4684  016166  032777  000020  176740  4$:   BIT     #BIT04,@SWR     ;PLU TESTED?
 4685  016174  001005                        BNE     5$              ;BRANCH IF NO
 4686  016176  005077  162216                CLR     @DRCSR          ;MUST PULSE DRCSR TO GET INTERRUPT.
 4687  016202  052777  000101  162210        BIS     #BIT06+BIT00,@DRCSR     ;SET PLU INTERRUPT AND REQUEST A
 4688
 4689
 4690  016210  000002                  5$:   RTI                     ;RETURN INLINE-NEXT TEST
 4691  016212  001000                  ICOUNT: 1000
 4692  016214  000000                  SCOPEF: 0                     ;COUNT LOCATION FOR ITERATION LOOP
 4693
 4694
 4695                          ;******************************************************************
 4696                          .SBTTL  OUTPUT CHARACTER TRAP ROUTINE
 4697                          ;******************************************************************
 4698
 4699  016216  105777  162212  OUTCHR: TSTB    @TPS            ;WAIT FOR RDY
 4700  016222  100375                        BPL     OUTCHR          ;
 4701  016224  110077  162206                MOVB    R0,@TPB         ;PRINT CHAR
```

```
4702  016230  000002                          RTI                     ;RETURN
4703
4704                                  ;******************************************************************
4705                                  .SBTTL POWER FAIL ROUTINE
4706                                  ;******************************************************************
4707
4708  016232  010046                  PFAIL:  MOV     R0,-(6)         ;SAVE REGISTER ON STACK
4709  016234  010146                          MOV     R1,-(6)         ;WHEN POWERING DOWN
4710  016236  010246                          MOV     R2,-(6)
4711  016240  010346                          MOV     R3,-(6)
4712  016242  010446                          MOV     R4,-(6)
4713  016244  010546                          MOV     R5,-(6)
4714  016246  013746  000024                  MOV     24,-(6)
4715  016252  010637  016330                  MOV     SP,PWRFLG       ;STORE STACK POSITION, POWER FAIL FLAG
4716  016256  012737  016266  000024          MOV     #PWRUP,24
4717  016264  000000                          HALT                    ;HALT ON POWER DOWN NORMAL
4718  016266  013706  016330          PWRUP:  MOV     PWRFLG,SP       ;RESTORE STACK POINTER
4719  016272  005000                          CLR     R0              ;DELAY ON POWER UP FOR TTY SAKE
4720  016274  005300                  1$:     DEC     R0
4721  016276  100776                          BMI     1$              ;DELAY AWHILE
4722  016300  104001  017104                  TYPE    ,PWR            ;TELL POWER FAILED
4723  016304  012637  000024                  MOV     (6)+,24         ;RESTORE POWER FAIL VECTOR
4724  016310  012605                          MOV     (6)+,R5         ;RESTORE R5,
4725  016312  012604                          MOV     (6)+,R4         ;THRU R0
4726  016314  012603                          MOV     (6)+,R3
4727  016316  012602                          MOV     (6)+,R2
4728  016320  012601                          MOV     (6)+,R1
4729  016322  012600                          MOV     (6)+,R0
4730  016324  000002                          RTI                     ;RETURN TO MAIN LINE
4731  016326  000000                  FIN:    0
4732  016330  000000                  PWRFLG: 0                       ;SAVE SP DURING POWER FAIL
4733
4734
4735
4736                                  ;******************************************************************
4737                                  .SBTTL  MEMORY SIZER AND RELOCATION
4738                                  .SBTTL SIZER TESTS FOR MEMORY STARTING AT BOTTOM
4739                                  ;IF RXDP CHAIN MODE DON'T SIZE MEMORY AND DEFAULT TO 4K TEST.
4740                                  ;******************************************************************
4741
4742  016332  012737  017776  000572  RELOC:  MOV     #17776,MEMORY   ;INIT SIZE OF MEMORY TO 4K
4743
4744  016340  005737  000540                  TST     @#CHAIN         ;RXDP CHAIN MODE?
4745  016344  001004                          BNE     1$              ;BRANCH IF YES
4746  016346  032777  001000  176560          BIT     #1000,@SWR      ;USER WANT CORE EXPANSION?
4747  016354  001401                          BEQ     DET4            ;BRANCH IF YES
4748  016356  000207                  1$:     RTS     PC              ; RETURN AND TEST 4K ONLY
4749
4750  016360  012737  016432  000004  DET4:   MOV     #DET2,4         ;TRAP VECTOR SETUP
4751  016366  012737  000340  000006          MOV     #340,6          ;TRAP STATUS SETUP
4752  016374  005537  037770          EIGHT:  ADC     @#37770         ;CHECK FOR 8K
4753  016400  005537  057770          TWELVE: ADC     @#57770         ;CHECK FOR 12K
4754  016404  005537  077770          SXTEEN: ADC     @#077770        ;CHECK FOR 16K
4755  016410  005537  117770          TWENTY: ADC     @#117770        ;CHECK FOR 20K
4756  016414  005537  137770          TWOFOR: ADC     @#137770        ;CHECK FOR 24K
4757  016420  005537  157770          TWOEIG: ADC     @#157770        ;CHECK FOR 28K
```

```
4758  016424  012702  157770                  MOV     #157770,R2      ;SET R2= 28K-6 ADDRESS
4759  016430  000407                          BR      TSTMEM          ;GO TEST 28K ADDRESS
4760  016432  012601                  DET2:   MOV     (SP)+,R1        ;GET TRAP PC
4761  016434  024141                          CMP     -(R1),-(R1)     ;POINT TO TOP OF MEMORY ADDRESS+2
4762  016436  014102                          MOV     -(R1),R2        ;SET R2=LAST ADDRESS-6
4763  016440  005726                          TST     (SP)+           ;DISCARD TRAP STATUS WORD
4764  016442  023702  016372                  CMP     EIGHT-2,R2      ;IF 4K R2 POINTS TO EIGHT-2
4765
4766  016446  001544                          BEQ     DET3            ;4K
4767                                  ;*****************************************************************
4768                                  .SBTTL MEMORY ADDRESS TEST OF ALL MEMORY BANKS IF RELOCATION SELECTED
4769                                  ;REGISTER USAGE;
4770                                  ;R0=CURRENT ADDRESS
4771                                  ;R1=FIRST ADDRESS
4772                                  ;R2=LAST ADDRESS-6
4773                                  ;R4=CURRENT ADDRESS+2
4774                                  ;R3=LAST ADDRESS
4775                                  ;*****************************************************************
4776  016450  010203                  TSTMEM: MOV     R2,R3           ;GET LAST ADDRESS-6
4777  016452  062703  000006                  ADD     #6,R3           ;MAKE IT LAST ADDRESS
4778  016456  010337  000572                  MOV     R3,MEMORY       ;SAVE MAXIMUM MEMORY ON SYSTEM
4779  016462  012700  020000                  MOV     #20000,R0       ;R0= FIRST ADDRESS
4780  016466  010001                          MOV     R0,R1           ;SAVE FIRST ADDRESS FOR END OF TEST CHECK
4781
4782                                  ;FILL MEMORY FROM 20000 TO TOP WITH EACH LOCATIONS ADDRESS
4783  016470  010004                          MOV     R0,R4           ;SET R4=FIRST ADDRESS
4784
4785  016472  010024                  1$:     MOV     R0,(R4)+        ;STORE AN ADDRESS
4786  016474  010400                          MOV     R4,R0           ;SET R0=NEW CURRENT ADDRESS
4787  016476  020003                          CMP     R0,R3           ;DONE ALL OF AVAIL. MEMORY?
4788  016500  101774                          BLOS    1$              ;BRANCH IF NO
4789
4790                                  ;TEST EACH MEMORY LOCATION IN DESCENDING ORDER
4791
4792  016502  020340                  2$:     CMP     R3,-(R0)        ;IS THE DATA OK?
4793  016504  001401                          BEQ     .+4             ;BRANCH IF YES
4794  016506  104000                          HLT                     ;*ERROR* MEMORY ADDRESS TEST FAILED;
4795                                                                  ;SAVR0=LOCATION FAILED
4796                                                                  ;SAVR3=EXPECTED DATA (= SAVR0)
4797  016510  162703  000002                  SUB     #2,R3           ;DECREMENT DATA/ADDRESS
4798  016514  020103                          CMP     R1,R3           ;DONE TO 20000?
4799  016516  101771                          BLOS    2$              ;BRANCH IF NO
4800
4801
4802                                  ;STORE PROCESSOR INSTRUCTION TEST IN EACH 4K MEMORY BANK
4803
4804  016520  023702  016376                  CMP     EIGHT+2,R2
4805  016524  001442                          BEQ     STRT8           ;8K
4806  016526  023702  016402                  CMP     TWELVE+2,R2
4807  016532  001434                          BEQ     STRT12          ;12K
4808  016534  023702  016406                  CMP     SXTEEN+2,R2
4809  016540  001426                          BEQ     STRT16          ;16K
4810  016542  023702  016412                  CMP     TWENTY+2,R2
4811  016546  001420                          BEQ     STRT20          ;20K
4812  016550  023702  016416                  CMP     TWOFOR+2,R2
4813  016554  001412                          BEQ     STRT24          ;24K
```

```
4814  016556  000406                        BR      STRT28          ;28K
4815  016560  005000                MOVE:   CLR     R0              ;SET UP MAIN CORE CURRENT
4816  016562  012021                1$:     MOV     (0)+,(1)+       ;MOVE WORD
4817  016564  020027  015214               CMP     R0,#ENDREL      ;MOVE COMPLETE?
4818  016570  001374                       BNE     1$              ;MOVE ANOTHER WORD
4819  016572  000207                       RTS     PC              ;MOVE COMPLETE
4820  016574  004737  016640        STRT28: JSR     PC,XFER28       ;START 28K TRANSFER
4821  016600  000450                       BR      MOD24           ;START 24K MODIFY
4822  016602  004737  016650        STRT24: JSR     PC,XFER24       ;START 24K TRANSFER
4823  016606  000450                       BR      MOD20           ;START 20K MODIFY
4824  016610  004737  016660        STRT20: JSR     PC,XFER20       ;START 20K TRANSFER
4825  016614  000450                       BR      MOD16           ;START 16K MODIFY
4826  016616  004737  016670        STRT16: JSR     PC,XFER16       ;START 16K TRANSFER
4827  016622  000450                       BR      MOD12           ;START 12K MODIFY
4828  016624  004737  016700        STRT12: JSR     PC,XFER12       ;START 12K TRANSFER
4829  016630  000450                       BR      MOD8            ;START 8K MODIFY
4830  016632  004737  016710        STRT8:  JSR     PC,XFER8        ;START 8K TRANSFER
4831  016636  000450                       BR      MOD4            ;START 4K MODIFY
4832  016640  012701  140000        XFER28: MOV     #140000,R1      ;SET UP MOVE START LOCATION
4833  016644  004737  016560               JSR     PC,MOVE         ;GO TO MOVE SUBROUTINE
4834  016650  012701  120000        XFER24: MOV     #120000,R1
4835  016654  004737  016560               JSR     PC,MOVE
4836  016660  012701  100000        XFER20: MOV     #100000,R1
4837  016664  004737  016560               JSR     PC,MOVE
4838  016670  012701  060000        XFER16: MOV     #60000,R1
4839  016674  004737  016560               JSR     PC,MOVE
4840  016700  012701  040000        XFER12: MOV     #40000,R1
4841  016704  004737  016560               JSR     PC,MOVE
4842  016710  012701  020000        XFER8:  MOV     #20000,R1
4843  016714  004737  016560               JSR     PC,MOVE
4844  016720  000207                       RTS     PC              ;RETURN FROM TRANSFERS
4845  016722  012737  143600  135212 MOD24:  MOV     #BEGIN+140000,TRPA+120002
4846  016730  012737  123600  115212 MOD20:  MOV     #BEGIN+120000,TRPA+100002
4847  016736  012737  103600  075212 MOD16:  MOV     #BEGIN+100000,TRPA+60002
4848  016744  012737  063600  055212 MOD12:  MOV     #BEGIN+60000,TRPA+40002
4849  016752  012737  043600  035212 MOD8:   MOV     #BEGIN+40000,TRPA+20002 ;STORE JMP TO 12K TEST
4850  016760                       MOD4:
4851  016760  000207                DET3:   RTS     PC              ;RETURN FROM MODIFY
4852
4853
4854
4855                                ;*****************************************************************
4856                                .SBTTL UTILITY #1-PRINT REGISTETERS AT TIME OF ERROR
4857                                ;*****************************************************************
4858
4859
4860
4861  016762  012706  017474        PNTABL: MOV     #STACK,SP       ;SETUP STACK POOINTER
4862  016766  012737  000011  017026        MOV     #ELAST-EFIRST/2,PNTMP   ;SIZE OF TABLE
4863  016774  012705  000550               MOV     #EFIRST,R5      ;BEGIN OF TABLE
4864  017000  012503                1$:     MOV     (R5)+,R3        ;GET AN ENTRY
4865  017002  004737  016012               JSR     PC,TYPOCT       ;PRINT OCTAL
4866  017006  104001  017146               TYPE    ,CR             ;PRINT CR
4867  017012  005337  017026               DEC     PNTMP           ;DONE WHOLE TABLE?
4868  017016  001370                       BNE     1$              ;BRANCH IF NO
4869  017020  000000                       HALT                    ;PNTABL ROUTINE NORMAL HALT
```

```
4870  017022  000137  000602                    JMP     @#START         ;IN CASE OF CONTINUE
4871  017026  000000                    PNTMP:  0                       ;TEMP STORAGE FOR PNTABL
4872
4873                                    ;*****************************************************************
4874                                    .SBTTL ASR 33 PAPER TAPE PUNCH ROUTINE
4875                                    ;*****************************************************************
4876
4877                                    ;       PUNCHES A TEST TAPE IN A BINARY COUNT PATTERN  FROM 0-377
4879                                    ;AND 50 FRAMES OF BLANK TAPE. THE PROGRAM IS CONTINOUS RUNNING.
4879                                    ;       THE RESULTANT TAPE CAN BE USED AS AN INPUT TAPE FOR THE ASR 33
4880                                    ;LOW SPEED READER TEST.
4881
4882  017030  012706  017474           ASRUTL: MOV     #STACK,SP       ;SETUP STACK
4883  017034  005001                            CLR     R1              ;INIT DATA PATTERN
4884  017036  012700  000017                    MOV     #15.,R0         ;INIT 50 BLANK FRAME COUNT.
4885  017042  105777  161366           1$:     TSTB    @TPS            ;IS PRINTER READY?
4886  017046  100375                            BPL     1$              ;BRANCH IF NO
4887  017050  005077  161362                    CLR     @TPB            ;PRINT NULL/BLANK FRAME
4888  017054  005300                            DEC     R0              ;DONE 50 FRAMES?
4889  017056  003371                            BGT     1$              ;BRANCH IF NO
4890  017060  105777  161350           2$:     TSTB    @TPS            ;IS PRINTER READY?
4891  017064  100375                            BPL     2$              ;BRANCH IF NO
4892  017066  110177  161344                    MOVB    R1,@TPB         ;PRINT 0-377 DATA PATTERN
4893  017072  105201                            INCB    R1              ;UPDATE DATA PATTERN
4894  017074  120127  000377                    CMPB    R1,#377         ;DONE DATA PATTERN?
4895  017100  101767                            BLOS    2$              ;BR IF NO
4896  017102  000752                            BR      ASRUTL          ;KEEP  DOING THIS UNTIL  HALT LINE SET
4897
4898
4899                                    ;*****************************************************************
4900                                    .SBTTL MESSAGE STORAGE
4901                                    ;*****************************************************************
4902
4903  017104  053520  022522     000   PWR:    .ASCIZ 'PWR%'
4904  017111     045  041520  000075   PCE:    .ASCIZ '%PC='
4905  017116  050040  053523  000075   CCE:    .ASCIZ ' PSW='
4906  017124  042515  047515  054522   MAXM:   .ASCIZ  'MEMORY= '
      017132  020075     000
4907  017135     040  042522  052524   SCPE:   .ASCIZ ' RETURN='
      017142  047122  000075
4908  017146  000045                    CR:     .ASCIZ '%'
4909  017150  051040  042530  051122   RXS:    .ASCIZ  ' RXERROR='
      017156  051117  000075
4910  017162  050045  051501  036523   PAS:    .ASCIZ  '%PASS='
      017170     000
4911  017171     040  051105  047522   ERR:    .ASCIZ  ' ERROR='
      017176  036522     000
4912  017201     040  044524  042515   PTIM:   .ASCIZ  ' TIME='
      017206  000075
4913  017210  042045  045526  044101   MREV:   .ASCIZ '%DVKAH-A%'
      017216  040455  000045
4914
4915                                    .EVEN
4916
4923
4924
```

```
4925
4926            017474                  .=17474
4927
4928  017474   000000                   STACK:  0               ;BEGIN OF STACK
4929                                    .SBTTL  STACK BUFFER AREA
4930            000001                          .END
```

DVKAHA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
A       015100          1807    1815    1832    1840    1848    1856    1865    1872    1881    1889    1916    1925    1957
                        1967    2022    2040    2049    2467    2522    2604    2612    2621    2634    2654    2673    2681
                        2689    2697    2705    2713    2722    2729    2738    2749    2761    2770    2800    2810    3206
                        3289    3297    3306    3318    3372    4381#   4382    4384    4386
ACT     015512          4536#
ASHCO   014136          3908#
ASHC1   014156          3919#
ASHO    014070          3877    3884#
ASH1    014110          3895#
ASKASR  001064          1063    1077#
ASKCON  001040          1060#
ASKEIS  001512          1179    1197#
ASKHSP  001130          1083    1088#
ASKHSR  001152          1089    1095#
ASKLP   001550          1198    1208    1214#
ASKPLU  001434          1111    1177#
ASKRX   001204          1108#
ASLO    012570          3486#
ASL67   012624          3505#
ASRINR  001736           785    1275#
ASRUTL  017030           871    4882#   4896
B       015070          2031    2232    2239    2246    2257    2266    2273    2274    2285    2303    2316    2325    2333
                        2394    2401    2408    2419    2428    2435    2436    2448    2466    2475    2484    2492    2505
                        3014    3021    3029    3036    3047    3056    3063    3064    3134    3141    3148    3158    3167
                        3174    3175    3187    3205    3336    3337    3344    3345    3352    3353    4376#   4377    4380
BANKO   015214          4428    4446#
BEGIN   003600          1264    1748#   4515    4526    4845    4846    4847    4848    4849
BITBO   004546          2009#
BITO    004530          1998#
BITOO = 000001           715#    740    1110    1117    1391    1655    4687
BIT01 = 000002           714#    740    1110    1120    1193    1651
BIT02 = 000004           713#   1062
BIT03 = 000010           712#    748    1197    1405    3876
BIT04 = 000020           711#    740     748    1178    1412    1530    4684
BIT05 = 000040           710#   1193    1214    1482    1654    1690
BIT06 = 000100           709#   1082    1391    1507    4687
BIT07 = 000200           708#    748
BIT08 = 000400           707#
BIT09 = 001000           706#   1244    4511
BIT10 = 002000           705#   1262
BIT11 = 004000           704#   4671
BIT12 = 010000           703#   1242    4519
BIT13 = 020000           702#
BIT14 = 040000           701#   1503    4669
BIT15 = 100000           700#
BREND   003730          1799#
BRXX    003624          1760#
C       015112          1935    2049*   2072*   2092*   2111*   2131*   2151*   2173*   2195*   2273*   2274    2302*   2303*
                        2304    2435*   2436    2465*   2466*   2467    2643*   2645    2651*   2655    2780    2874*   2894*
                        2916*   2936*   2956*   2978*   3000*   3063*   3064    3174*   3175    3204*   3205*   3206    3327*
                        3329    3360    3361*   3363    3369*   3373    4389#   4390    4392
CCE     017116          4579    4905#
CHAIN   000540           939#   1002*   1007*   4513    4744
CHECK   000530           935#    999*   1412*   1651*   1655*   4456
CHECKM= 000023           740#   4455
CLINCT  002604          1218*   1429    1431*   1438    1440*   1443*   1458#
```

DVKAHA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
CLOCK   003570          794  1727#
CLRTBT  015426         4501  4509  4512  4514  4517#
CPUTST  001720         1261# 1263
CR      017146         1237  1286  1288  4491  4585  4866  4908#
CTCHLT  000170          806# 1289
CURPAT  002600         1219* 1432  1434* 1436* 1450  1453* 1456#
CWAIT = 000230          748# 4299  4300
D       015132         2062* 2082* 2101* 2121* 2141* 2162* 2184* 2844* 2854* 2864* 2884* 2903* 2926*
                       2946* 2967* 2989* 4399#
DELAY   002304         1079* 1101* 1359  1364  1371#
DET2    016432         4750  4760#
DET3    016760         4766  4851#
DET4    016360         4747  4750#
DIVO    014260         3956#
DRCSR   000420          881# 1193* 1378  1391* 1401  1405* 4686* 4687*
DRIBUF  000424          883# 1383  1407
DROBUF  000422          882# 1192* 1390* 1413*
DRVECA  000426          884# 1185
EAESRT  015174         4308  4422#
EFIRST  000550          953# 4862  4863
EIGHT   016374         4752# 4764  4804
ELAST   000572          963# 4862
ENDPAS  015266         4468#
ENDREL  015214         4429# 4817
ERR     017171         4482  4911#
ERROR   000526          934#  998* 4483  4568*
ESTART  000666         1006  1010# 4284  4541  4594
FADDO   014326         3992#
FDIVO   014526         4077#
FIN     016326         4731#
FISLVL  000246          846#
FISVEC  000244          845#
FMULO   014460         4052#
FSTACK  014602         3992  4024  4052  4077  4095#
FSUBO   014412         4024#
HIGHA   014606         4005  4032  4058  4083  4097#
HLT   = 104000          736#  846  1047  1056  1092  1099  1134  1143  1147  1153  1157  1209  1222
                       1277  1294  1311  1332  1338  1358  1380  1386  1403  1410  1426  1484  1494
                       1505  1537  1556  1560  1595  1599  1622  1627  1668  1765  1768  1770  1772
                       1776  1782  1789  1795  1799  1809  1817  1834  1842  1850  1858  1867  1874
                       1883  1891  1906  1919  1928  1937  1946  1960  1970  1980  1990  2001  2012
                       2024  2033  2042  2051  2065  2075  2085  2095  2104  2114  2124  2134  2144
                       2154  2165  2176  2187  2198  2210  2224  2234  2241  2248  2260  2268  2276
                       2288  2297  2306  2319  2327  2336  2348  2357  2366  2375  2384  2396  2403
                       2410  2422  2430  2438  2451  2460  2469  2478  2486  2495  2508  2517  2525
                       2538  2547  2556  2565  2574  2584  2594  2606  2614  2623  2637  2647  2657
                       2664  2675  2683  2691  2699  2707  2715  2724  2731  2742  2751  2764  2773
                       2782  2791  2803  2813  2823  2833  2847  2857  2867  2877  2887  2897  2906
                       2909  2919  2929  2939  2949  2959  2970  2981  2992  3003  3016  3023  3071
                       3038  3050  3058  3066  3078  3087  3096  3105  3114  3124  3136  3143  3150
                       3161  3169  3177  3190  3199  3208  3220  3229  3238  3247  3256  3266  3269
                       3279  3291  3299  3308  3321  3331  3339  3347  3355  3365  3375  3388  3398
                       3411  3413  3415  3417  3432  3434  3437  3445  3447  3450  3459  3461  3464
                       3473  3475  3478  3493  3497  3512  3516  3531  3535  3550  3554  3565  3567
                       3570  3579  3581  3584  3637  3640  3663  3666  3675  3678  3702  3707  3739
                       3745  3762  3774  3786  3798  3839  3860  3869  3889  3901  3913  3924  3929
```

```
                    3944    3949    3964    3969    4004    4009    4036    4062    4087    4149    4156    4163    4169
                    4177    4458    4794
HLTMES  015540       772    4554#
HPCSR   000444       893#   1090    1094*   1356
HPDBR   000446       894#   1354*
HPOUT   002210      1353#   1366
HPOUTR  002224       791    1356#
HPOUT1  002214      1354#   1367
HPOUT2  002270      1362    1365#
HRCSR   000440       891#   1097    1102*   1330    1342*   1359*   1364*
HRDBR   000442       892#   1333    1336
HSPDAT  002300      1078*   1353*   1354    1365*   1369#
HSPINH  000546       948#   1088
HSRDAT  002206      1077*   1336    1339*   1341*   1345#
HSRINH  000544       947#   1095
HSRINR  002134       789    1330#
HSRIN1  002172      1341#
HSRIN2  002200      1335    1340    1342#
ICOUNT  016212      1010*   1754*   3613*   4123*   4292*   4307*   4673    4691#
INTCNT  002302      1360*   1361    1363*   1370#
KBBDAT  002060      1080*   1084*   1279    1292    1296*   1298*   1302#
LINKER  015476      4520    4524    4533#
LOCATE  000522       932#   1751*
LOWA    014610      4007    4034    4060    4085    4098#
LP    = 000000       726#   1416
LPCSR   000450       895#   1220    1224*   1424
LPDBR   000452       896#   1223*   1441*   1444*   1450*
LPEX    002576      1442    1451    1454#
LPINTR  002424       866    1424#
LPSO    000542       946#   1218    1429    1438
MARKSP  015132      3867*   4398#
MARKO   013766      3849#
MATH    014612      3878    4089    4123#
MATHLP  014634      4128#   4178
MATHND  014774      4180#
MAXM    017124      1234    4906#
MEMORY  000572       965#   1235    4742*   4778*
MES     015740      4560    4608#
MOD12   016744      4827    4848#
MOD16   016736      4825    4847#
MOD20   016730      4823    4846#
MOD24   016722      4821    4845#
MOD4    016760      4831    4850#
MOD8    016752      4829    4849#
MOVE    016560      4815#   4833    4835    4837    4839    4841    4843
MREV    017210      1008    4913#
MTPSO   013514      3696#
MTPS27  013554      3734#
MULO    014216      3938#
N     = 017222      4918#   4919
NODEVI  000600       972#   1035    1177
NOP   = 000240       735#
OLDRS   000536       938#   3852*   3865
OUTCHR  016216       766    4699#   4700
PAS     017162      4479    4910#
PASR    000414       861     871#
```

```
PASS    000524          933#   997*  4462*  4480
PASSNO  000576          967#  1014*  1241*  1246*  4447
PASSN1  000574          966#  1013*  4446*  4447   4463*
PC    =%000007         1142*  1152*  1230*  1236*  1287*  1504*  1536*  1555*  1559*  1594*  1598*  1621*  1697*
                       1748   2207   3384*  3389*  3395*  3409*  3623*  3641*  3651*  3652*  3667*  3679*  3851
                       3858   4409*  4419*  4481*  4484*  4487*  4490*  4537*  4578*  4591*  4584*  4660*  4748*
                       4819*  4820*  4822*  4824*  4826*  4828*  4830*  4833*  4835*  4837*  4839*  4841*  4843*
                       4844*  4851*  4865*
PCE     017111         4576   4904#
PERROR  015572         4568#
PFAIL   016232          769   4422   4708#
PLUA    002352         1186   1401#
PLUB    002306         1188   1378#
PLUDAT  000534          937#  1191*  1383   1389*  1390   1407   1411*  1413
PNTABL  016762          870   4861#
PNTCHR= 000004          739#  4615   4617   4647
PNTMP   017026         4862*  4867*  4871#
PREG    000410          870#
PS    = 177776          721#
PTIM    017201         4488   4912#
PWR     017104         4722   4903#
PWRFLG  016330         4281   4283*  4715*  4718   4732#
PWRUP   016266         4716   4718#
P40   = ****** U       1019   1023   1252   1256   3690   4470   4475   4493   4497
REF   = 013240         3644#
REFF    013240         3621*  3622*  3624   3628   3638   3642#  3644   3649*  3650*  3653   3656   3664   3676
RELOC   016332         1230   4742#
RETURN  000532          936#   996*  1750*  4452*  4583   4676   4681*
RK    = ****** U        838    921   1165   1699   1820
RCTALL  013120         3623   3628#
RCTBE   013300         3651   3656#
RCTBO   013410         3652   3669#
RCTEN1  013512         3655   3685#
RXCON   002644         1125*  1487   1488#  1502*  1542*  1566*  1579*  1602*  1637*  1647*
RXCSR   000454          902#  1141*  1151*  1477   1503*  1507*  1535*  1554*  1593*  1618*  1672   1690
RXDAT   000472          909#  1114*  1508*  1539   1540*  1578*  1625   1629*  1648*
RXDB    000456          903#  1145   1155   1479   1539*  1558*  1562*  1597*  1601*  1624
RXFRST  002742         1125   1502   1523#  1566   1647
RXHCNT  000504          914#  1689*  1693*
RXINER  002646         1481   1492#
RXINT   002606          850   1127   1477#
RXLCNT  000502          913#  1671*  1674*  1688*  1692*
RXLVL   000266          851#
RXRD    003262         1602   1616#
RXRSEC  003206         1579   1591#  1637
RXRTRY  000520          920#  1160*  1492   1501*  1634*
RXS     017150         4485   4909#
RXSA    000476          911#  1124*  1510*  1558   1567*  1568   1580*  1597   1638*  1639   1645*  1657*
RXSAB   000466          907#  1124   1510   1580   1645   1657
RXSOFT  000516          919#  1000*  1113*  1500*  4486
RXTA    000500          912#  1123*  1509*  1562   1571   1575*  1577*  1601   1642   1646*  1658*
RXTAB   000470          908#  1123   1509   1571   1577   1642   1646
RXUNO   000462          905#  1115*  1119*  1129   1133*  1528
RXUN1   000464          906#  1116*  1122*  1149   1525
RXUTT   000474          910#  1159*  1523   1527*  1530*  1552   1591   1649   1652*  1654*
RXVEC   000264          850#   904
```

```
RXVECP  000460          904#  1126
RXWERR  003566         1676   1695   1697#
RXWRET  003562         1673   1691   1696#
RXWSEC  003042         1542   1552#
RXWTDN  003522         1142   1152   1504   1688#
RXWTTR  003500         1536   1555   1559   1594   1598   1621   1671#
R0     =%000000        1108*  1109*  1110*  1126*  1127*  1128*  1185*  1186*  1187*  1188*  1189*  1190*  1191
                       1192   1201*  1202*  1203*  1206*  1207*  1748*  1749*  1750   1751   1806*  1814*  1831*
                       1839*  1847*  1855*  1863*  1879*  1887*  1903*  1904   1915*  1924*  1933*  1942*  1956*
                       1966*  1976*  1985*  1998*  1999   2021*  2030*  2039*  2048*  2061*  2071*  2081*  2091*
                       2100*  2110*  2120*  2130*  2140*  2150*  2160*  2171*  2182*  2193*  2207*  2208*  2209
                       2216*  2217   2257*  2258   2284*  2285*  2286   2315*  2316*  2317   2419*  2420   2447*
                       2448*  2449   2474*  2475*  2476   2504*  2505*  2506   2603*  2611*  2619*  2632*  2642*
                       2652*  2672*  2680*  2688*  2696*  2704*  2712*  2720*  2736*  2747*  2760*  2769*  2778*
                       2787*  2799*  2809*  2819*  2828*  2843*  2853*  2863*  2873*  2883*  2893*  2902*  2915*
                       2925*  2935*  2945*  2955*  2965*  2976*  2987*  2998*  3047*  3048   3158*  3159   3186*
                       3187*  3188   3288*  3296*  3304*  3316*  3326*  3336*  3344*  3352*  3360*  3370*  3486*
                       3488*  3495   3524*  3526*  3533   3696*  3697*  3703*  3704*  3705   3757*  3759*  3760
                       3769*  3771   3781*  3783   3793*  3795*  3796   3849*  3863   3868   3884*  3886*  3895*
                       3897*  3899   3908*  3910*  3919*  3921*  3927   3938*  3939*  3945   3957*  3959*  3967
                       4025*  4026   4027*  4028   4029   4030   4124*  4128*  4130   4147   4154   4161   4165
                       4166   4167   4417*  4418*  4453*  4454*  4455*  4456*  4471*  4494*  4536*  4537   4555
                       4558*  4559*  4561*  4610*  4612   4614*  4616*  4620*  4636   4642*  4645*  4646*  4648*
                       4650*  4652*  4657*  4701   4708   4719*  4720*  4729*  4779*  4780   4783   4785   4786*
                       4787   4792   4815*  4817   4884*  4888*
R1     =%000001        1204*  1616   1624*  1625   1632*  1864*  1880*  1888*  2217*  2218   2620*  2653*  2721*
                       2737*  2748*  3305*  3371*  3756*  3758*  3760   3770*  3771*  3772   3909*  3920*  3925
                       3947   3956*  3965   3992*  3993*  3994*  3995*  3996*  3999*  4024*  4026*  4028*  4029*
                       4030*  4031*  4052*  4053*  4054*  4055*  4056*  4057*  4077*  4078*  4079*  4080*  4081*
                       4082*  4130*  4131*  4132*  4133*  4134*  4135*  4136*  4137*  4138*  4139*  4140*  4141*
                       4142*  4143*  4144*  4145*  4146*  4147   4151*  4152*  4153*  4154   4158*  4160*  4161
                       4165*  4166*  4167   4171   4174*  4175   4298*  4299*  4300   4554   4556*  4557   4558
                       4562*  4608   4609*  4610   4621*  4637   4641*  4653*  4658*  4709   4728*  4760*  4761
                       4762   4780*  4798   4832*  4834*  4836*  4838*  4840*  4842*  4883*  4892   4893*  4894
R2     =%000002        1038*  1062   1082   1108   1117   1120   1178   1197   1214   1477*  1478   1480   1482
                       1534*  1541*  1552*  1553*  1554   1591*  1592*  1593   1617*  1630*  4638   4710   4727*
                       4758*  4762*  4764   4776   4804   4806   4808   4810   4812
R3     =%000003        1039*  1043   1045   1053   1054   1064   1081   1094   1101   1224   1235*  1285*  2218*
                       2220   4171*  4172*  4173*  4175   4415*  4416*  4480*  4483*  4486*  4489*  4577*  4590*
                       4583*  4639   4644*  4649*  4651*  4659*  4711   4726*  4776*  4777*  4778   4787   4792
                       4797*  4798   4864*
R4     =%000004        1040*  1085   1102   2220*  2221   4125*  4174   4413*  4414*  4640   4712   4725*  4783*
                       4785*  4786
R5     =%000005        2221*  2222   3850*  3851*  3852   3853   3856*  3859*  3865   4126*  4178*  4411*  4412*
                       4713   4724*  4863*  4864
SAVPC   000566          961#  4573*  4577
SAVPS   000570          962#  4574*  4580
SAVR0   000550          954#  4555*  4561   4620   4636*  4657
SAVR1   000552          955#  4554*  4562   4621   4637*  4658
SAVR2   000554          956#  4638*
SAVR3   000556          957#  4639*  4659
SAVR4   000560          958#  4640*
SAVR5   000562          959#
SAVSP   000564          960#  4571*  4572*
SCOPE = 104400          738#  1810   1818   1835   1843   1851   1859   1868   1875   1884   1892   1907   1920
                       1929   1938   1947   1961   1971   1981   1991   2002   2013   2025   2034   2043   2052
```

```
                    2066    2076    2086    2096    2105    2115    2125    2135    2145    2155    2166    2177    2188
                    2199    2212    2225    2235    2242    2249    2261    2269    2277    2289    2298    2307    2320
                    2328    2337    2349    2358    2367    2376    2385    2397    2404    2411    2423    2431    2439
                    2452    2461    2470    2479    2487    2496    2509    2518    2526    2539    2548    2557    2566
                    2575    2585    2595    2607    2615    2624    2638    2648    2658    2665    2676    2684    2692
                    2700    2708    2716    2725    2732    2743    2752    2765    2774    2793    2792    2804    2814
                    2824    2834    2848    2858    2868    2878    2889    2898    2910    2920    2930    2940    2950
                    2960    2971    2982    2993    3004    3017    3024    3032    3039    3051    3059    3067    3079
                    3088    3097    3106    3115    3125    3137    3144    3151    3162    3170    3178    3191    3200
                    3209    3221    3230    3239    3248    3257    3270    3280    3292    3300    3309    3322    3332
                    3340    3348    3356    3366    3376    3390    3400    3418    3425    3438    3451    3465    3479
                    3498    3517    3536    3555    3571    3585    3637    3663    3675    3685    3709    3747    3763
                    3775    3787    3799    3840    3870    3902    3930    3950    3970    4010    4037    4063    4088
                    4180    4306
SCOPEC  016114       775    4669#
SCOPEF  016214      1015*   4673    4675*   4680*   4692#
SCPE    017135      4582    4907#
SETLK   001200      1096    1103#
SETLP   001556      1217#
SETPLU  001450      1185#
SETREL  001622      1215    1230#
SETRX   001216      1113#
SOB     013624      3756#
SOLPAT  002602      1217*   1445    1447*   1452*   1453    1457#
SP     =%000006      995*   1011*   1037*   1051*   1285    1616*   1632    1696*   2216    2222    3386    3396    3849
                    3853*   3854*   3855*   3856    3857*   3858*   3863    3867    3868*   4517*   4525*   4556    4571
                    4622*   4676*   4681    4715    4718*   4760    4763    4861*   4882*
SPACE = ****** U    3711    3800    3815    4182    4316
SR     = 000176      749#
SRXCSR  000510       916#   1478#
SRXDB   000512       917#
SRXES   000506       915#   1479#
SRXSB   000514       918#
STACK   017474       995    1011    1037    1051    4861    4882    4928#
START   000602       835     995#    996    4870
START2  001000      1043#
STRT12  016624      4807    4828#
STRT16  016616      4809    4826#
STRT20  016610      4811    4824#
STRT24  016602      4813    4822#
STRT28  016574      4814    4820#
STRT8   016632      4805    4830#
SUBR1   015136      4409#
SUBR2   015140      4410#   4412
SUBR3   015144      4412#   4414
SUBR4   015152      4414#   4416
SUBR5   015160      4416#   4418
SUBR6   015166      3409    4418#
SWABO   012666      3524#
SWAB67  012722      3543#
SWR     015134      1032    1034*   1038    1242    1244    1262    3876    4298    4403#   4453    4468    4511    4519
                    4569    4586    4592    4669    4671    4684    4746
SWREG   000176       749     818#   1034
SXTEEN  016404      4754#   4808
SXTOO   013720      3793#
SXT61   013740      3834#
```

```
TEMP    015122          1916* 1917  1925* 1926  1934* 1943* 1944  1955* 1957* 1958  1965* 1967* 1968
                        1975* 1977* 1978  1986* 1987* 1988  2009* 2010  2020* 2022* 2029* 2031* 2038*
                        2040* 2047* 2060* 2063  2070* 2073  2080* 2083  2090* 2093  2099* 2102  2109*
                        2112  2119* 2122  2129* 2132  2139* 2142  2149* 2152  2159* 2163  2170* 2174
                        2181* 2185  2192* 2196  2265* 2266  2293* 2294* 2295  2324* 2325* 2332* 2333*
                        2334  2344* 2345* 2346  2353* 2354* 2355  2362* 2363* 2364  2371* 2372* 2373
                        2380* 2381* 2382  2427* 2428  2456* 2457* 2458  2483* 2484* 2491* 2492* 2493
                        2513* 2514* 2515  2521* 2522* 2523  2534* 2535* 2536  2543* 2544* 2545  2552*
                        2553* 2554  2561* 2562* 2563  2570* 2571* 2572  2579* 2581* 2582  2589* 2591*
                        2592  2633* 2634* 2635  2644* 2654* 2761* 2762  2770* 2771  2779* 2788* 2789
                        2798* 2800* 2801  2808* 2810* 2811  2818* 2820* 2821  2829* 2830* 2831  2842*
                        2845  2852* 2855  2862* 2865  2872* 2875  2882* 2885  2892* 2895  2901* 2904
                        2907  2914* 2917  2924* 2927  2934* 2937  2944* 2947  2954* 2957  2964* 2968
                        2975* 2979  2986* 2990  2997* 3001  3055* 3056  3074* 3075* 3076  3083* 3084*
                        3085  3092* 3093* 3094  3101* 3102* 3103  3110* 3111* 3112  3119* 3120* 3121*
                        3122  3166* 3167  3195* 3196* 3197  3216* 3217* 3218  3225* 3226* 3227  3234*
                        3235* 3236  3243* 3244* 3245  3252* 3253* 3254  3261* 3263* 3264  3267  3274*
                        3276* 3277  3317* 3318* 3319  3328* 3372* 3429* 3430* 3435  3442* 3443* 3448
                        3456* 3457* 3462  3470* 3471* 3476  3505* 3507* 3514  3543* 3545* 3552  3562*
                        3563* 3568  3576* 3577* 3582  3740* 3741* 3742* 3743  3782* 3783* 3794  3834*
                        3836* 3837  4393# 4394  4396  4397
TEST    013242          3628* 3629* 3630* 3631* 3632* 3633* 3634* 3635* 3636* 3638  3643# 3656* 3657*
                        3658* 3659* 3660* 3661* 3662* 3664  3669* 3670* 3671* 3672* 3673* 3674* 3676
TIME    003576          1103* 1727* 1729# 4489
TJSR1   012300          3385# 3386
TJSR2   012302          3384  3386#
TJSR3   012314          3385  3390# 3396
TKB     000432           888# 1281  1283  1290  1292
TKS     000430           887# 1081* 1085* 1275  1299*
TPB     000436           890# 1319* 4701* 4887* 4892*
TPBDAT  002130          1060* 1313  1318* 1319  1322# 4500*
TPBNUL  002132          1061* 1315* 1317* 1323#
TPS     000434           889# 1043* 1045  1053* 1054  1064* 1309  4699  4885  4890
TRPA    015210          4428# 4845* 4846* 4847* 4848* 4849*
TRPB    015532          1012* 4288  4507  4518* 4523  4542#
TSROT   013076          3622# 3625
TSROT2  013252          3650# 3654
TSRT2A  013244          3626  3649#
TSTMEM  016450          4759  4776#
TWELVE  016400          4753# 4806
TWENTY  016410          4755# 4810
TWOEIG  016420          4757#
TWOFOR  016414          4756# 4812
TYOUTR  002062           787  1309#
TYOUTI  002120          1316  1319#
TYPE  = 104001           737# 1008  1234  1237  1286  1288  4479  4482  4485  4488  4491  4576  4579
                        4582  4585  4722  4866
TYPOCT  016012          1236  1287  4481  4484  4487  4490  4578  4581  4584  4636# 4865
WAITST  015030          4298#
WAITX   015066          4289  4308#
WAIT4   015056          4301  4306#
WAIT5   015014          4282  4288#
XFER12  016700          4828  4840#
XFER16  016670          4826  4838#
XFER20  016660          4824  4836#
XFER24  016650          4822  4834#
```

```
XFER28  016640          4820    4832#
XFER8   016710          4830    4842#
XORO    013650          3769#
XOR67   013674          3781#
XX      = 000000         752#   1302    1322    1345    1369
XXXXX   = 000000         753#   1488
YESRT   015534          4521    4543#
YESTR   015376          4448    4469    4507#
YESTR1  015470          4526#   4534
$ENDAD  015516           781    1005    4537#   4588
.       = 017476         754#    756#    760     762#    763     765#    768#    771#    774#    780#    782#    784#    793#
                         802#    817#    827#    832     844#    845     849#    860#    863#    969#   1046    1055    1146
                        1156    1260    1276    1310    1331    1337    1357    1379    1385    1402    1409    1425    1483
                        1626    1764    1767    1769    1771    1775    1781    1788    1794    1798    1808    1816    1833
                        1841    1849    1857    1866    1873    1882    1890    1905    1918    1927    1936    1945    1959
                        1969    1979    1989    2000    2011    2023    2032    2041    2050    2064    2074    2084    2094
                        2103    2113    2123    2133    2143    2153    2164    2175    2186    2197    2223    2233    2240
                        2247    2259    2267    2275    2287    2296    2305    2318    2326    2335    2347    2356    2365
                        2374    2383    2395    2402    2409    2421    2429    2437    2450    2459    2468    2477    2485
                        2494    2507    2516    2524    2537    2546    2555    2564    2573    2583    2593    2605    2613
                        2622    2636    2646    2656    2663    2674    2682    2690    2698    2706    2714    2723    2730
                        2741    2750    2763    2772    2781    2790    2802    2812    2822    2832    2846    2856    2866
                        2876    2886    2896    2905    2908    2918    2928    2938    2948    2958    2969    2980    2991
                        3002    3015    3022    3030    3037    3049    3057    3065    3077    3086    3095    3104    3113
                        3123    3135    3142    3149    3160    3168    3176    3189    3198    3207    3219    3228    3237
                        3246    3255    3265    3266    3278    3290    3298    3307    3320    3330    3338    3346    3354
                        3364    3374    3387    3397    3410    3412    3414    3416    3431    3433    3436    3444    3446
                        3449    3458    3460    3463    3472    3474    3477    3492    3496    3511    3515    3530    3534
                        3549    3553    3564    3566    3569    3578    3580    3583    3637    3639    3663    3665    3675
                        3677    3701    3706    3738    3744    3761    3773    3785    3797    3838    3850    3857    3888
                        3900    3912    3923    3928    3943    3948    3963    3969    4003    4008    4035    4061    4086
                        4148    4155    4162    4168    4176    4380#   4384#   4392#   4396#   4457    4589    4793    4918
                        4926#
```

DVKAHA.P11 CROSS REFERENCE TABLE -- MACRO NAMES

```
STARS    691#   696   698   801   804   813   815   828   830   856   858   862   865   873   875
          898   900   942   994   950   952   984   992  1028  1030  1181  1183  1226  1228  1266
         1272  1304  1307  1325  1328  1347  1350  1373  1376  1396  1399  1419  1422  1464  1475
         1517  1520  1545  1549  1584  1588  1607  1613  1663  1669  1680  1686  1723  1726  1731
         1734  1740  1745  1756  1758  1802  1804  1895  1897  1910  1912  1951  1953  1994  1996
         2005  2007  2016  2018  2056  2058  2203  2205  2228  2230  2253  2255  2280  2282  2311
         2313  2340  2342  2389  2391  2415  2417  2443  2445  2500  2502  2530  2532  2599  2601
         2628  2630  2668  2670  2755  2757  2794  2796  2838  2840  3009  3011  3043  3045  3070
         3072  3129  3131  3154  3156  3182  3184  3212  3214  3284  3286  3312  3314  3379  3381
         3403  3405  3421  3423  3482  3484  3501  3503  3520  3522  3539  3541  3557  3559  3589
         3591  3615  3617  3645  3647  3688  3692  3730  3732  3752  3754  3765  3767  3777  3779
         3789  3791  3830  3832  3842  3847  3873  3875  3880  3882  3891  3893  3904  3906  3915
         3917  3932  3934  3952  3954  3973  3989  4012  4022  4039  4050  4065  4075  4105  4120
         4277  4279  4294  4296  4311  4313  4432  4436  4441  4443  4464  4466  4503  4505  4529
         4531  4547  4552  4564  4566  4598  4605  4628  4635  4662  4666  4695  4697  4704  4706
         4736  4740  4767  4775  4855  4857  4873  4875  4899  4901
TNCV     3593#  3637  3663  3675
```

```
ADC     1693  2162  2173  2581  4158  4752  4753  4754  4755  4756  4757
ADCB    2967  2978  3263
ADD     1567  1638  1692  1696  2208  2505  2514  2522  3851  3858  4165  4572  4622  4777
ASH     3886  3897
ASHC    3910  3921
ASL     3488  3507  4173
ASLB    3577
ASR     4172
ASRB    3563
BCC     1761  1781  3489  3637  3663  3675  3698  3887  3898  3911  3922  4003
BCS     1767  1779  3416  3431  3444  3458  3472  3508  3527  3546  3564  3578  3735  3943  3960
        4560
BEQ     1004  1006  1046  1063  1111  1130  1150  1243  1282  1291  1293  1335  1337  1366  1385
        1409  1433  1524  1626  1643  1650  1771  1793  1808  1816  1833  1841  1849  1857  1866
        1873  1882  1890  1905  1918  1927  1936  1945  1959  1969  1979  1989  2000  2023  2032
        2041  2050  2064  2074  2084  2094  2103  2113  2123  2133  2143  2153  2164  2175  2186
        2197  2223  2233  2240  2247  2259  2267  2275  2287  2296  2305  2318  2326  2335  2347
        2356  2365  2374  2383  2395  2402  2409  2421  2429  2437  2450  2459  2468  2477  2485
        2494  2507  2516  2524  2537  2546  2555  2564  2573  2583  2593  2605  2613  2622  2636
        2646  2656  2674  2682  2690  2698  2706  2714  2723  2730  2741  2750  2763  2772  2781
        2790  2802  2812  2822  2832  2846  2856  2866  2876  2886  2896  2905  2908  2918  2928
        2938  2948  2958  2969  2980  2991  3002  3015  3022  3030  3037  3049  3057  3065  3077
        3086  3095  3104  3113  3123  3135  3142  3149  3160  3168  3176  3189  3198  3207  3219
        3228  3237  3246  3255  3265  3268  3278  3290  3298  3307  3320  3330  3338  3346  3354
        3364  3374  3387  3397  3412  3436  3449  3463  3477  3491  3496  3510  3515  3529  3534
        3553  3569  3583  3639  3665  3677  3706  3737  3744  3761  3773  3785  3797  3838  3866
        3877  3888  3900  3912  3928  3942  3948  3962  3968  4001  4008  4035  4061  4086  4148
        4155  4162  4168  4176  4282  4301  4457  4509  4534  4593  4611  4674  4747  4766  4793
        4805  4807  4809  4811  4813
BGE     1430  1787
BGT     1439  1570  1641  1786  1798  4889
BHI     1774
BHIS    4448
BIC     1064  1110  1359  1405  1957  1967  1977  1987  2285  2294  2303  2448  2457  2466  2634
        2644  2654  3704  3742  4174  4299  4455  4456
BICB    2800  2810  2820  2830  3187  3196  3205  3318  3328  3372
BIS     1043  1053  1081  1085  1094  1102  1224  1364  1412  1503  1507  1553  1592  1651  1654
        1655  4646  4687
BIT     1045  1054  1062  1082  1117  1120  1178  1197  1214  1242  1244  1262  1482  1690  1999
        3876  4511  4519  4569  4592  4669  4671  4684  4746
BITB    2010
BLE     1788
BLO     1780
BLOS    1775  4788  4799  4895
BLT     1446  1493  1569  1640  1794  2663
BMI     1146  1156  1221  1276  1310  1331  1357  1379  1402  1425  1481  1673  1764  3410  3701
        3923  3963  4000  4289  4469  4587  4721
BNE     1055  1083  1089  1096  1118  1121  1132  1179  1198  1215  1245  1263  1280  1284  1297
        1314  1340  1362  1483  1526  1529  1572  1675  1691  1694  1763  2011  3548  3625  3654
        3700  3864  3926  3946  3966  4006  4033  4059  4084  4512  4514  4520  4570  4589  4613
        4654  4670  4672  4685  4745  4818  4868
BPL     1091  1098  1316  1773  1792  3492  3511  3530  3549  3637  3663  3675  3738  3940  4524
        4700  4886  4891
BR       832   861  1033  1138  1208  1260  1367  1435  1437  1442  1448  1451  1573  1576  1653
        1656  1676  1695  3385  3626  3637  3655  3663  3675  4089  4308  4501  4618  4677  4759
        4814  4821  4823  4825  4827  4829  4831  4896
```

```
BVC     1762  3446  3490  3566  3580  3637  3663  3675  3699
BVS     1769  1785  3414  3433  3460  3474  3509  3528  3547  3637  3663  3675  3736  3941  3961
        4002
CCC     3394  3408  3428  3455  3506  3561
CLC     1778  3885  3896  4643
CLN     1797  3794
CLR      997   998   999  1000  1002  1010  1012  1013  1014  1015  1060  1078  1079  1080  1103
        1113  1114  1115  1116  1133  1159  1160  1187  1189  1206  1241  1250  1353  1363  1443
        1508  1527  1578  1634  1648  1652  1671  1688  2062  2072  2345  2362  2371  2504  2513
        2535  2552  2561  3101  3110  3119  3234  3243  3613  3781  3782  3834  3908  3920  3993
        4283  4452  4463  4500  4517  4522  4642  4648  4680  4686  4719  4815  4883  4887
CLRB    2844  2854  2864  2874  3075  3217  3362
CLV     1784
CLZ     1791
CMP     1005  1361  1383  1407  1429  1432  1438  1445  1568  1571  1639  1642  1807  1815  1832
        1840  1848  1856  1865  1872  1881  1889  1917  1926  1935  1944  1958  1968  1978  1988
        2142  2152  2222  2232  2239  2246  2258  2266  2274  2286  2295  2304  2317  2355  2364
        2373  2382  2394  2401  2408  2420  2428  2436  2449  2458  2467  2476  2506  2515  2523
        2545  2554  2563  2572  2604  2612  2621  2635  2645  2655  2855  2865  2907  2947  2957
        2968  2979  2990  3076  3085  3094  3103  3112  3122  3264  3337  3345  3363  3373  3435
        3448  3462  3476  3495  3514  3533  3552  3568  3582  3624  3638  3653  3664  3676  3705
        3760  3772  3837  3863  3865  3899  3925  3927  3945  3947  3967  4005  4032  4083  4085
        4147  4154  4161  4167  4175  4447  4575  4588  4673  4761  4764  4787  4792  4798  4804
        4806  4808  4810  4812  4817
CMPB    1131  1281  1283  1292  1336  1492  1625  1904  2662  2673  2681  2689  2697  2705  2713
        2722  2729  2738  2749  2762  2771  2780  2789  2801  2811  2821  2831  3014  3021  3029
        3036  3048  3056  3064  3134  3141  3148  3159  3167  3175  3188  3197  3206  3227  3236
        3245  3254  3289  3297  3306  3319  3329  3353  3386  3396  4300  4612  4894
COM     1109  1389  2082  2092  2354  2544  4152  4454  4518
COMB    2884  2894  3084  3093  3226
DEC     2121  2131  2372  2562  4027  4139  4140  4141  4142  4143  4144  4145  4146  4720  4867
        4888
DECB    1315  2926  2936  3111  3244  3276  4653
DIV     3959
EMT      736   737
FADD    3999
FDIV    4082
FMUL    4057
FSUB    4031
HALT     755   760   764   806   972  4544  4590  4717  4869
INC     1007  1084  1119  1122  1299  1342  1360  1411  1431  1434  1440  1452  1500  1674  1727
        2101  2111  2363  2553  3622  3650  3758  4131  4132  4133  4134  4135  4136  4137  4138
        4153  4446  4462  4568  4675
INCB    1296  1318  1339  1365  1501  1540  1629  2903  2916  3102  3235  4893
IOT      739
JMP      835   866   870   871  1264  1289  1487  2209  3878  4284  4428  4515  4541  4594  4870
JSR     1142  1152  1230  1236  1287  1504  1536  1555  1559  1594  1598  1621  3384  3395  3409
        3623  3651  3652  4412  4414  4416  4418  4481  4484  4487  4490  4537  4578  4581  4584
        4820  4822  4824  4826  4828  4830  4833  4835  4837  4839  4841  4843  4865
MFPS    3703  3741  4471
MOV      995   996  1011  1031  1034  1035  1036  1037  1038  1039  1040  1051  1061  1077  1101
        1108  1123  1124  1125  1126  1127  1128  1141  1151  1177  1185  1186  1188  1190  1191
        1192  1193  1201  1202  1203  1207  1217  1218  1219  1223  1235  1246  1249  1285  1298
        1317  1319  1341  1354  1390  1391  1413  1436  1441  1444  1447  1450  1453  1477  1478
        1479  1502  1509  1510  1530  1534  1535  1542  1552  1554  1558  1562  1566  1575  1577
        1579  1580  1591  1593  1597  1601  1602  1616  1617  1618  1632  1637  1645  1646  1647
```

	1657	1658	1689	1748	1750	1751	1754	1806	1814	1831	1839	1847	1855	1863	1864
	1879	1880	1887	1888	1915	1916	1924	1925	1933	1934	1942	1943	1955	1956	1965
	1966	1975	1976	1985	1986	1998	2020	2021	2029	2030	2038	2039	2047	2048	2060
	2061	2070	2071	2080	2081	2090	2091	2099	2100	2109	2110	2119	2120	2129	2130
	2139	2140	2149	2150	2159	2160	2170	2171	2181	2182	2192	2193	2207	2216	2217
	2218	2220	2221	2257	2265	2273	2284	2293	2302	2315	2324	2332	2344	2353	2380
	2419	2427	2435	2447	2456	2465	2474	2483	2491	2521	2534	2543	2570	2579	2589
	2603	2611	2619	2620	2632	2633	2642	2643	2651	2652	2653	2672	2680	2688	2696
	2704	2712	2720	2721	2736	2737	2747	2748	2760	2769	2778	2787	2798	2799	2808
	2809	2818	2819	2828	2829	2842	2843	2852	2853	2862	2863	2872	2873	2882	2883
	2892	2893	2901	2902	2914	2915	2924	2925	2934	2935	2944	2945	2954	2955	2964
	2965	2975	2976	2986	2987	2997	2998	3074	3083	3092	3186	3195	3204	3216	3225
	3252	3261	3274	3288	3296	3304	3305	3316	3317	3326	3327	3336	3344	3352	3360
	3361	3369	3370	3371	3429	3442	3456	3470	3486	3505	3524	3543	3562	3576	3621
	3628	3649	3656	3696	3740	3756	3757	3769	3770	3793	3849	3850	3852	3853	3854
	3855	3856	3857	3867	3868	3884	3895	3909	3919	3938	3956	3957	3992	3994	3995
	3996	4024	4025	4026	4028	4029	4030	4052	4053	4054	4055	4056	4077	4078	4079
	4080	4081	4123	4124	4125	4126	4130	4171	4292	4298	4307	4422	4423	4453	4480
	4483	4486	4489	4521	4525	4526	4535	4536	4554	4555	4556	4558	4561	4562	4571
	4573	4574	4577	4580	4583	4609	4614	4616	4620	4621	4636	4637	4638	4639	4640
	4657	4658	4659	4676	4681	4708	4709	4710	4711	4712	4713	4714	4715	4716	4718
	4723	4724	4725	4726	4727	4728	4729	4742	4750	4751	4758	4760	4762	4776	4778
	4779	4780	4783	4785	4786	4816	4832	4834	4836	4838	4840	4842	4845	4846	4847
	4848	4849	4861	4862	4863	4864	4882	4884							
MOVB	1539	1624	1903	2009	2761	2770	2779	2788	3047	3055	3063	3120	3158	3166	3174
	4610	4641	4701	4892											
MTPS	1020	1253	3697	3734	4472	4494									
MUL	1204	3939													
NEG	2141	2151	2381	2571	4151										
NEGB	2946	2956	3121	3253											
NOP	834	2211	4538	4539	4540										
RESET	1044	1052													
ROL	3629	3634	3635	3636	4644	4645	4649	4650	4651	4652					
ROLB	3457	3471	3660	3661	3662	3672	3673	3674							
ROR	3630	3631	3632	3633	4559										
RORB	3430	3443	3657	3658	3659	3669	3670	3671							
RTI	807	1300	1320	1343	1355	1394	1414	1454	1511	1543	1581	1603	1659	1728	4527
	4595	4623	4690	4702	4730										
RTS	1697	3389	3641	3667	3679	3859	4409	4411	4413	4415	4417	4419	4660	4748	4819
	4844	4851													
RTT	4543														
SBC	2184	2195	2591	4160											
SBCB	2989	3000													
SCC	1760	3441	3469	3487	3525	3544	3575	4410							
SEC	2161	2172	2183	2194	2580	2590	2966	2977	2988	2999	3262	3275	3958	3998	4157
	4159														
SEN	3835														
SEV	3997														
SOB	1541	1630	3759	4178											
SUB	1749	2022	2031	2040	2049	2316	2325	2333	2475	2484	2492	4128	4166	4797	
SWAB	3526	3545													
SXT	3795	3836													
TRAP	738														
TST	1032	1088	1090	1095	1097	1279	1378	1480	1523	1525	1528	1649	2063	2073	2083
	2093	2102	2112	2122	2132	2163	2174	2185	2196	2334	2346	2493	2536	2582	2592
	3001	3277	3399	3743	3784	3796	3965	4007	4034	4058	4060	4281	4288	4507	4513

DVKAHA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
        4523  4557  4586  4608  4744  4763
TSTB    1003  1129  1145  1149  1155  1220  1275  1290  1309  1313  1330  1333  1356  1401  1424
        1672  2845  2875  2885  2895  2904  2917  2927  2937  3218  3267  4468  4533  4699  4885
        4890
WAIT    1261  4302  4303  4304  4305
XOR     3771  3783
.ASCIZ  4903  4904  4905  4906  4907  4908  4909  4910  4911  4912  4913
.DSABL  1744
.ENABL   687   733  4434
.END    4930
.ENDC    842   928  1021  1026  1172  1254  1258  1459  1720  1830  3728  3749  3813  3828  4274
        4374  4473  4478  4495  4499  4921
.EVEN   4915
.IF      838   921  1019  1023  1165  1252  1256  1416  1699  1820  3690  3711  3800  3815  4182
        4316  4470  4475  4493  4497  4919
.LIST       7   684   686   729   746   761   837  3612  4044  4070  4922
.MACR   3593
.MACRO   691
.NLIST      5     6   683   685   718   742   836  3587  4042  4068  4917
.PAGE    680
.REPT      9   757
.SBTTL   803   814   829   857   864   874   899   943   951   970   985  1029  1182  1227  1267
        1268  1269  1270  1305  1326  1348  1374  1420  1465  1724  1732  1733  1741  1742  2227
        3874  4286  4312  4433  4435  4442  4504  4530  4548  4629  4663  4696  4705  4737  4738
        4768  4856  4874  4900  4929
.TITLE   689
.WORD    783  4095  4096  4097  4098
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

```
*,DVKAHA/CRF=DVKAHA.P11
RUN-TIME:  14 28 5 SECONDS
RUN-TIME RATIO:  220/47=4.6
CORE USED:  12K  (23 PAGES)
```

Spooler runtime 15 Seconds, 65 KCS, 402 disk reads, 3 disk writes, 101 pages