PDP11/34

MEMORY MANAGEMENT BASIC LOGIC

MD-11-DFKTA-A

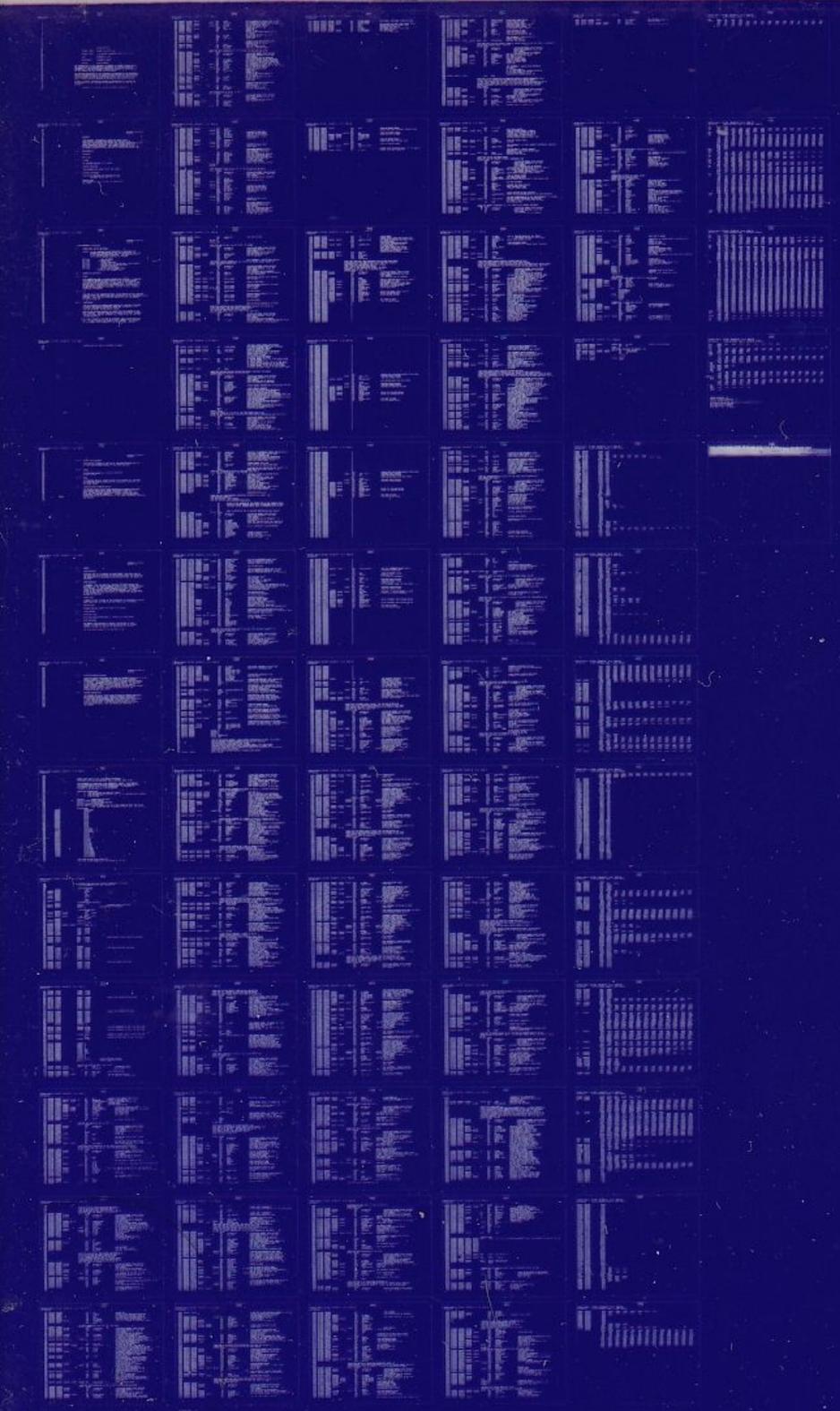
EP-DFKTA-A-DL-A

OCT 1976

FICHE 1 OF 1

digital

Made in U.S.A.



```
.REM#
```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DFKTA-A-D
PRODUCT NAME:	11/34 MEMORY MANAGEMENT BASIC LOGIC TEST
DATE:	DECEMBER 21,1975
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	GLENN JOHNSON

1.0 ABSTRACT

THIS PROGRAM INCREMENTALLY TESTS THE BASIC LOGIC FUNCTIONS OF THE MEMORY MANAGEMENT FOR THE PDP-11/34. THEY FULLY TEST RELOCATION, DIRECT AND INDIRECT ADDRESSING OF THE MEMORY MANAGEMENT REGISTERS, AND CORRECT OPERATION OF ALL THE BITS IN THE REGISTERS. THE VARIOUS ABORTS ARE TESTED, AS IS PROPER "LOCKING" AND "UNLOCKING" OF THE ERROR TRACKING LOGIC.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11/34

2.2 STORAGE

THE PROGRAM REQUIRES 4K OF MEMORY.

3.0 LOADING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABS LOADER.

4.0 STARTING PROCEDURE

LOAD SWITCH REGISTER WITH DESIRED SETTING.
(SOFTWARE SWITCH REG. LOC. = 176)

START AT 200.
THE PROGRAM WILL RING THE BELL ON COMPLETION OF A PASS.

5.0 OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

@@NOTE: IF NO HARDWARE SWITCH REAGISTER IS AVAILABLE THE PROGRAM WILL AUTOMATICALLY USE THE CONTENTS OF LOC. 176 AS THE SOFTWARE SWITCH REGISTER. THE USER SHOULD SET THIS LOCATION BEFORE STARTING THE PROGRAM.

```
BIT 15=1        -- HALT ON ERROR
BIT 14=1        -- SCOPE LOOP
BIT 13=1        -- INHIBIT PRINTOUT
BIT 12=1        -- INHIBIT BELL AT END OF PASS, TYPE ASTERICK
BIT 12=0        -- RING BELL AT END OF EACH PASS
BIT 11=1        -- INHIBIT ITERATIONS
BIT 10=1       -- HALT AT END OF CURRENT TEST
                WITH NEXT TEST NUMBER IN R0.
```

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUBTEST. IT RECORDS THE STARTING ADDRESS OF EACH SUBTEST AS IT IS BEING ENTERED. IF A SCOPE LOOP IS REQUESTED, IT WILL JUMP TO THE START OF THE SUBTEST THAT THE SCOPE LOOP IS REQUESTED FOR. IF SCOPE LOOP IS NOT REQUESTED, THERE WILL BE 1024 ITERATIONS ON THAT SUBTEST BEFORE THE NEXT SUBTEST IS ENTERED. SWITCH 11 ON A 1 INHIBITS ITERATION OF SUBTESTS.

5.2.2 HLT

THIS EMT CALLS THE SUBROUTINE PRINT, WHICH PRINTS OUT THE LOCATION COUNTER AT THE TIME OF FAILURE AND THE CONTENTS OF THE PROCESSOR STATUS REGISTER. NOTE THAT THE LOCATION COUNTER WILL BE THE ADDRESS OF THE HLT PLUS TWO.

5.2.3 TRAPCATCHER

THIS IS A SERIES OF INSTRUCTIONS STARTING AT LOCATION 0 DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS TO THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.

EACH VECTOR ENTRANCE ADDRESS IS LOADED WITH THE ADDRESS OF THE NEXT LOCATION. THE NEXT LOCATION IS LOADED WITH A HALT (000000). THUS AN ILLEGAL TRAP OR INTERRUPT WILL CAUSE A HALT AT THE TRAP LOCATION PLUS TWO.

IF A HALT OCCURS IN THE TRAP OR INTERRUPT AREA EXAMINE REGISTER SIX. IT WILL CONTAIN THE CURRENT STACK ADDRESS. THE CONTENTS OF THE CURRENT STACK ADDRESS IS THE VALUE OF THE LOCATION

```
140                                     COUNTER WHEN THE TRAP OR INTERRUPT OCCURRED.
141
142
```

F01

5.2.4 EMTSRV (EMT DECODER)

THIS ROUTINE DECODES ALL EMT CALLS, INCLUDING PATCHES AND THE HLT CALL WHICH PASSES CONTROL TO THE PRINT ROUTINE.

5.2.5 CLRALL

THIS ROUTINE CLEARS ALL THE PAR'S AND PDR'S.
AS WELL AS SR0.

5.2.6 RWALL

THIS ROUTINE MAPS ALL PAGES TO BANK 0 BY CLEARING ALL THE PAR'S.
ALL PAGES ARE MADE 4K READ-WRITE BY LOADING ALL THE PDR'S WITH
THE VALUE 77406.

5.3 PROGRAM AND/OR OPERATOR ACTION

THE PROGRAM FIRST CHECKS THOSE PROPERTIES OF MEMORY MANAGEMENT
WHICH CAN BE TESTED WITH MEMORY MANAGEMENT TURNED OFF.
THEN, DESTINATION ONLY RELOCATION IS USED TO SHOW THAT BASIC
RELOCATION IS WORKING CORRECTLY. FINALLY, FULL RELOCATION IS
ENABLED AND MISCELLANEOUS ASPECTS OF THE MEMORY MANAGEMENT OPERATION
ARE CHECKED.

6.0 ERRORS

6.1 ERROR PRINTOUT

PRINTOUTS ARE IN A STANDARD TWO-WORD FORMAT. THE FIRST WORD IS THE OCTAL VALUE OF THE PC+2 OF THE DETECTED ERROR. THE SECOND IS THE CONTENTS OF THE PROCESSOR STATUS REGISTER WHEN THE ERROR WAS DETECTED.

6.2 ERROR RECOVERY

IN GENERAL, TEST FAILURES WILL PRINTOUT AN ERROR MESSAGE AND CONTINUE. IF THE "HALT ON ERROR" SWITCH IS SET, HITTING CONTINUE WILL RECOVER. IF THE PROGRAM HANGS UP IN A LOOP, THE ERROR IS LIKELY TO BE A SIGNAL WHICH WAS NEVER RECEIVED. IF A HALT OCCURS IN THE TRAP AND VECTOR AREA THE PROGRAM MUST BE RESTARTED, IF THE PROGRAM HALTS IN THE MAIN FLOW, CONSULT THE LISTING IF NO MESSAGE IS TYPED OUT.

6.3 BRANCH SELF

A BRANCH TO SELF IS USED IN THIS DIAGNOSTIC TO INDICATED A FAILURE WHEN A HALT OR A HLT WORD TRAP CALL COULD LEAD TO PROBLEM.

7.0 RESTRICTIONS

PROGRAM MUST BE LOADED INTO LOWER 4K OF MEMORY.

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

EACH PASS TAKES APPROXIMATELY 1 MINUTE WITH CORE MEMORY.

8.2 STACK POINTERS

THE KERNEL STACK POINTER IS USUALLY INITIALIZED TO 1000. HOWEVER, IN CERTAIN TESTS IT MAY BE INITIALIZED TO A LOWER ADDRESS (VIRTUAL) TO MAKE UP FOR RELOCATION OF THE BANK.

THE USER STACK POINTER IS INITIALIZED TO 400.

8.4 EXECUTION ORDER CHECKING

SINCE THE MEMORY MANAGEMENT MAY CAUSE AN INCORRECT FETCH IF IT IS NOT WORKING CORRECTLY, THE ORDER OF EXECUTION OF ALL SUBTESTS IS CHECKED. THE SCOPE ROUTINE, WHEN IT CHANGES FROM ONE SUBTEST TO THE NEXT, INCREMENTS A COUNTER CALLED TESTCT. AT THE START OF EACH SUBTEST, THIS COUNTER IS CHECKED FOR THE CORRECT VALUE FOR THAT SUBTEST. IF TESTS ARE NOT EXECUTED IN THE CORRECT ORDER, TESTCT WILL NOT CONTAIN THE EXPECTED VALUE, AND AN ERROR PRINTOUT WILL OCCUR.

9.0 PROGRAM DESCRIPTION

THE PROGRAM INITIALLY TESTS THOSE FEATURES OF MEMORY MANAGEMENT WHICH CAN BE TESTED WITHOUT TURNING ON MEMORY MANAGEMENT. IT THEN USES THE MAINTENANCE MODE (DESTINATION ONLY RELOCATION) TO TEST TURNING MEMORY MANAGEMENT ON AND OFF AND TO FULLY CHECK OUT RELOCATION. ONCE RELOCATION HAS BEEN FULLY TESTED, FULL PAGING IS USED TO TEST THE REMAINING OPERATIONS OF THE OPTION.

*

```
                                ;BASIC LOGIC TEST OF THE 11/34 MEMORY MANAGEMENT
                                ;COPYRIGHT 1975 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

                                ;THIS PROGRAM IS A MODIFIED 11/40 DIAGNOSTIC, DBKTA.  THIS VERSION
                                ;HAS THE SOFTWARE SWITCH REGISTER CAPABILITIES AND HAS BEEN MODIFIED
                                ;TO ACCOUONT FOR AY 11/40 - 11/34 DIFFERENCES.  THIS PROGRAM IS
                                ;INTENDED TO BE RUN ONLY ON 11/34 PROCESSORS.

                                ;OPERATING INSTRUCTIONS
                                ;       1.  LOAD TEST USING THE ABSOLUTE LOADER
                                ;       2.  LOAD DESIRED SWITCH SETTING (LOC. 176 IS SOFTWARE SWR IF NEEDED)
                                ;       3.  START AT 200.

                                ;BIT15=1         CAUSES HALT ON ERROR
                                ;BIT14=1 CAUSES SCOPE LOOPING
                                ;BIT13=1         INHIBITS ERROR PRINTOUT
                                ;BIT11=1 INHIBITS ITERATIONS
                                ;BIT10=1 HALT AT END OF CURRENT TEST WITH TEST NUMBER OF NEXT TEST IN R0.
                                ;                TEST. PRESS CONTINUE TO ADVANCE TO NEXT TEST. (WITH BIT11=1)
                                ;DEFINITIONS
                                        SCOPE=TRAP
                                        NOP=240
                                        R0=%0
                                        R1=%1
                                        R2=%2
                                        R3=%3
                                        R4=%4
                                        R5=%5
                                        R6=%6
                                        R7=%7
                                        SP=%6
                                        PC=%7
                                        PS=177776
                                        STATUS=PS
                                        HLT=104006
                                        BIT0=1
                                        BIT1=2
                                        BIT2=4
                                        BIT3=10
                                        BIT4=20
                                        BIT5=40
                                        BIT6=100
                                        BIT7=200
                                        BIT8=400
                                        BIT9=1000
                                        BIT10=2000
                                        BIT11=4000
                                        BIT12=10000
                                        BIT13=20000
                                        BIT14=40000
                                        BIT15=100000

                                ;LOAD TRAP CATCHER INTO 0 THRU 777
                                ;LOAD EACH VECTOR ADDRESS WITH THE ADDRESS OF THE NEXT
```

```
                                        ;LOCATION, AND LOAD EACH LOCATION IMMEDIATELY FOLLOWING
                                        ;A VECTOR ADDRESS WITH A HALT INSTRUCTION
                                        ;LOAD VECTOR AREA
                000030                          .=30
000030  016334                                  EMTSRV
000032  000340                                  340
                000034                          .=34
000034  015544                                  $COPEC
000036  000000                                  0
                000046                          .=46
000046  015234                                  LOGIC
                                        ;SOFTWARE SWITCH REGISTER
                000174                          .=174
000174  000000                          DISPREG:       0              ;SOFTWARE DISPLAY REGISTER
000176  000000                          SWREG:  0               ;SOFTWARE SWITCH REGISTER
                                        ;LOAD STARTING AREA
                000200                          .=200
000200  000167  001000                          JMP     START
                000210                          .=210
000210  000167  015242                          JMP     TESTX
                                        ;LOAD DATA AREA
                000400                          .=400
000400  000000                          USTACK: 0
                001000                          .=.+376
001000  000000                          KSTACK: 0
001002  000000  000000  000000                  .WORD 0,0,0,0
001010  000000
001012  123456                          K123:   123456
001014  134567                          K134:   134567
001016  177564                          TCSR:   177564
001020  177566                          TDBR:   177566
001022  000000                          TEMP:   0
                                        ;
001024  177572                          SR0:    177572         ;MEMORY MANAG. STATUS REGISTER ADDRESSES
001026  177573                          SR0H:   177573
001030  177574                          SR1:    177574
001032  177576                          SR2:    177576
                                        ;
001034                                  ADRTAB:
001034  177600                          UPDR0:  177600         ;USER PAGE DESCRIPTOR REGISTERS
001036  177602                          UPDR1:  177602
001040  177604                          UPDR2:  177604
001042  177606                          UPDR3:  177606
001044  177610                          UPDR4:  177610
001046  177612                          UPDR5:  177612
001050  177614                          UPDR6:  177614
001052  177616                          UPDR7:  177616
                                        ;
001054  177640                          UPAR0:  177640         ;USER PAGE ADDRESS REGISTERS
001056  177642                          UPAR1:  177642
001060  177644                          UPAR2:  177644
001062  177646                          UPAR3:  177646
001064  177650                          UPAR4:  177650
```

```
                                UPAR5:  177652
                                UPAR6:  177654
                                UPAR7:  177656
                                ;
                                KPDR0:  172300          ;KERNEL PAGE DESCRIPTOR REGISTERS
                                KPDR1:  172302
                                KPDR2:  172304
                                KPDR3:  172306
                                KPDR4:  172310
                                KPDR5:  172312
                                KPDR6:  172314
                                KPDR7:  172316
                                ;
                                KPAR0:  172340          ;KERNEL PAGE ADDRESS REGISTERS
                                KPAR1:  172342
                                KPAR2:  172344
                                KPAR3:  172346
                                KPAR4:  172350
                                KPAR5:  172352
                                KPAR6:  172354
                                KPAR7:  172356
                                ADREND= .-2
                                ;
                                PDRTAB: 177600          ;STARTING ADDRESSES OF PDR'S FOR EACH MODE
                                PDREND: 172300
                                PARTAB: 177640          ;STARTING ADDRESSES OF PAR'S FOR EACH MODE
                                        172340
                                
                                STATAB: KPDR0           ;ADDRESS OF KERNEL TABLE OF PDR'S AND PAR'S
                                        0
                                        UPDR0           ;ADDRESS OF USER TABLE OF PDR'S AND PAR'S
                                STAEND: 140000
                                
                                
                                STAPNT: 0
                                PAGES:  0
                                SAVEA:  0
                                SAVEB:  0
                                KTVEC:  250
                                KTSTA:  252
                                PDRM2:  100361
                                FTITLE: 0
                                TESTCT: 0
                                BLOCKS: 0
                                SR:     177570          ;SWITCH REGISTER POINTER
                                DISPLAY:        0       ;DISPLAY REGISTER POINTER
                                
                                
                                ;SET UP FOR START OF BASIC LOGIC TESTS
                                START:  CLR     @#PS                    ;INITIALIZE STATUS
                                        MOV     #KSTACK,SP              ;SETUP KERNEL STACK
                                
                                        MOV     @#4,-(SP)       ;;SAVE ERROR VECTOR
                                        MOV     @#6,-(SP)
                                        MOV     #1$,4           ;;SET UP TIME OUT VECTOR
                                        TST     @SR             ;;TRY TO REFERENCE HARDWARE SWR
```

```
                                                BR      2$              ;;BRANCH IF NO TIMEOUT TRAP OCCURS
                                        1$:     MOV     #SWREG,SR       ;;POINT TO SOFTWARE SWR
                                                MOV     #DISPREG,DISPLAY;;POINT TO SOFTWARE DISPLAY REG
                                                CMP     (SP)+,(SP)+     ;;RESTORE STACK
                                        2$:     MOV     (SP)+,@#6       ;;RESTORE ERROR VECTOR
                                                MOV     (SP)+,@#4
                                                MOV     #2000,ICOUNT            ;INITIALIZE ITERATION COUNT
                                                MOV     #TEST1+2,RETURN         ;SETUP SCOPE AND ITERATION LOOP RETURN
                                                MOV     #1,TESTCT               ;INITIALIZE TEST COUNT
                                                TST     FTITLE                  ;DID TITLE PRINT
                                                BNE     TEST1+2                 ;YES, START TEST
                                                JSR     PC,TYPE                 ;NO, PRINT TITLE
                                                MTIT
                                                INC     FTITLE
                                                BR      .+4                     ;SKP SCOPE INSTRUCTION

                                ;SR0 AND SR1 SHOULD BE INITIALIZED TO 0
                                TEST1:  SCOPE
                                                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
                                                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
                                                1                               ;TEST NUMBER
                                                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
                                                RESET                           ;ISSUE INIT
                                                TST     @SR0                    ;CHECK SR0
                                                BEQ     .+4
                                                HLT                             ;SR0 WAS NOT INITIALIZED TO ZERO
                                                TST     @SR1                    ;CHECK SR1
                                                BEQ     .+4
                                                HLT                             ;SR1 WAS NOT INITIALIZED TO ZERO
                                                MOV     #10,ICOUNT              ;DROP ITERATION COUNT SINCE RESET IS USED

                                ;CHECK READ/WRITE PROPERTIES OF ALL BITS IN SR0 EXCEPT 0 AND 8
                                ;BY ROTATING A ONE THRU THE BIT POSITIONS BEING CHECKED
                                TEST2:  SCOPE
                                                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
                                                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
                                                2                               ;TEST NUMBER
                                                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
                                                TST     @SR0                    ;CHECK SR0 INITIALLY
                                                BEQ     .+6
                                                HLT                             ;SR0 NOT ZERO AT START OF TEST
                                                BR      EXIT2
                                                MOV     #1,R0                   ;R0 CONTAINS BIT INDICATING POSITION BEING TESTE
                                        LOOP2:  MOV     R0,R1
                                                MOV     R1,R2
                                                BIC     #401,R1         ;DON'T SET THE BIT IN SR0 IF IT'S BIT 0 OR BIT 8
                                                BIC     #17777,R2               ;CLEAR THE BIT IN R2 IF IT SHOULDN'T SET IN SR0
                                                MOV     R1,@SR0
                                                CMP     R2,@SR0                 ;CHECK SR0
                                                BEQ     .+4
                                                HLT                             ;SR0 INCORRECT WHEN VALUE IN R1
                                                                                ;WAS LOADED INTO IT
                                                ASL     R0
                                                BCC     LOOP2
                                                CLR     @SR0
                                        EXIT2:
```

```
                                        ;BITS 0-11 OF ALL PAR'S SHOULD BE READ/WRITE
                                        ;TEST BY ROTATING A BIT THRU EACH PAR
                                        ;ALSO SHOWS THAT OUTPUT PATHS FROM PAR'S ARE OK
                                        ;AND THAT EVERY PAR ADDRESS IS RESPONDED TO
001474  104400                  TEST3:  SCOPE
001476  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
001502  004767  014676                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
001506  000003                          3                               ;TEST NUMBER
001510  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
001512  012767  002000  014136          MOV     #2000,ICOUNT            ;RESTORE ICOUNT
001520  004767  013644                  JSR     %7,CLRALL               ;INITIALIZE MEMORY MANAG. REGISTERS
001524  012703  001140                  MOV     #PARTAB,R3              ;R3 POINTS TO TABLE OF PAR ADDRESSES
001530  012700  000002                  MOV     #2,R0                   ;R0 IS COUNTER OF STATES LEFT TO TEST
001534  012301                  LOOP3:  MOV     (R3)+,R1                ;PUT ADDRESS OF 1ST PAR IN SET IN R1
001536  012702  000010                  MOV     #10,R2                  ;R2 IS COUNTER OF PAR'S LEFT TO TEST IN SET
001542  012704  000001          LOOP3A: MOV     #1,R4                   ;R4 IS BIT OF PAR BEING TESTED
001546  010411          LOOP3B: MOV     R4,@R1                  ;SET BIT IN PAR
001550  020411                  CMP     R4,@R1                  ;CHECK PAR
001552  001401                  BEQ     .+4                     ;BRANCH IF OK
001554  104006                  HLT                             ;PAR WHOSE ADDRESS IS IN R1
                                                                ;FAILED WHEN THE VALUE IN R4
                                                                ;WAS LOADED INTO IT
001556  006304                  ASL     R4
001560  020427  010000          CMP     R4,#10000
001564  001370                  BNE     LOOP3B
001566  005011                  CLR     @R1
001570  005721                  TST     (R1)+                   ;MOVE POINTER
001572  077215                  SOB     R2,LOOP3A               ;TEST ALL PAR'S IN SET
001574  077021                  SOB     R0,LOOP3                ;TEST ALL 3 REGISTER SETS

                                        ;BITS 1-3, 8-14 OF ALL PDR'S SHOULD BE READ/WRITE
                                        ;BITS 0,4,5,7 AND 15 SHOULD ALWAYS BE ZERO
                                        ;BIT 6  SHOULD BE ZERO IF PDR IS WRITTEN
                                        ;ACTUAL CLEARING AND SETTING OF 6 TESTED LATER
                                        ;ALSO SHOWS THAT OUTPUT PATHS FROM PDR'S ARE OK
                                        ;AND THAT EVERY PDR ADDRESS IS RESPONDED TO
001576  104400                  TEST4:  SCOPE
001600  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
001604  004767  014574                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
001610  000004                          4                               ;TEST NUMBER
001612  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
001614  004767  013750                  JSR     %7,CLRALL               ;INITIALIZE MEMORY MANAG. REGISTERS
001620  012703  001134                  MOV     #PDRTAB,R3
001624  012301                  LOOP4:  MOV     (R3)+,R1                ;LOAD ADDRESS OF 1ST PDR IN STATE
001626  012702  000010                  MOV     #10,R2                  ;USE R2 AS A COUNTER OF PDR'S
                                                                        ;LEFT TO TEST
001632  012700  000001          LOOP4A: MOV     #1,R0                   ;SETUP R0 TO ROTATE A BIT THRU
001636  010005          LOOP4B: MOV     R0,R5
001640  046705  177324                  BIC     PDRM2,R5                ;R5 CONTAINS EXPECTED RESULTING CONTENTS OF PDR
001644  010011                  MOV     R0,@R1                  ;LOAD PDR
001646  021105                  CMP     @R1,R5                  ;CHECK RESULTING CONTENTS OF PDR
001650  001401                  BEQ     .+4
001652  104006                  HLT                             ;PDR WHOSE ADDRESS IS IN R1
                                                                ;WAS INCORRECT AFTER VALUE IN R0
                                                                ;WAS LOADED INTO IT
```

```
                                        ASL     R0              ;ROTATE BIT
                                        BCC     LOOP4B          ;BRANCH IF NOT DONE WITH THIS PDR
                                        CLR     @R1             ;IF DONE WITH THIS PDR, CLEAR IT
                                        TST     (R1)+           ;MOVE POINTER TO ADDRESS NEXT PDR
                                        SOB     R2,LOOP4A       ;TEST ALL PDR'S IN THIS GROUP
001136                                  CMP     R3,#PDREND      ;TEST ALL 2 GROUPS OF PDRS-USER, KERNEL
                                        BLE     LOOP4

                                ;NO DUAL ADDRESSING TEST FOR PAR'S AND PDR'S
                        TEST5:  SCOPE
001000                          MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
014476                          JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
000005                          5                               ;TEST NUMBER
104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
013452                          JSR     %7,CLRALL               ;CLEAR ALL PAR'S AND PDR'S
001034                          MOV     #ADRTAB,R1              ;R1 POINTS TO ADDRESS OF LOCATION
                                                                ;LOADED WITH 1 BIT SET IN EACH 4 BITS
001034                  LOP5AA: MOV     #ADRTAB,R2              ;R2 USED AS A POINTER TO CYCLE THRU
                                                                ;ALL OTHER ADDRESSES OF PAR/PDR PAIR'S TO
                                                                ;CHECK FOR DUAL ADDRESSING
000040                          MOV     #32.,R3                 ;R3 USED AS A COUNTER
010421  000000                  MOV     #10421,@(R1)            ;LOAD A PAR OR PDR - SET ONE BIT
                                                                ;IN EACH CHIP (4 BITS PER CHIP) IF R/W
                                LOP5B:  CMP     R2,R1           ;SKIP CHECKING THIS ADDRESS TO SEE IF
001406                          BEQ     CONT5                   ;IT'S A DUAL, SINCE IT WAS THE ONE LOADED
000000                          TST     @(R2)                   ;OTHERWISE, CHECK TO SEE IF THIS
                                                                ;REGISTER RESPONDED TO THE ADDRESS
                                                                ;OF THE ONE LOADED AS A DUAL
001403                          BEQ     CONT5                   ;BRANCH IF OK
104006                          HLT                             ;DUAL ADDRESSING - ADDRESS POINTED
                                                                ;TO BY R2 RESPONDED TO THE ADDRESS
                                                                ;POINTED TO BY R1 IN AT LEAST ONE
                                                                ;4 BIT SECTION (1 CHIP)
000000                          CLR     @(R2)                   ;REINITIALIZE FAULTY LOCATION
                                CONT5:  TST     (R2)+           ;MOVE POINTER R2
                                        SOB     R3,LOP5B        ;CHECK ALL PAR'S AND PDR'S
                                                                ;TO SEE IF THEY RESPONDED TO THE
                                                                ;ADDRESS POINTED TO BY R1
001132                          CMP     #ADREND,R1              ;HAVE ALL ADDRESSES BEEN CHECKED
                                                                ;FOR DUALS?
001402                          BEQ     DONE5A                  ;YES - GO TO NEXT TEST
005031                          CLR     @(R1)+                  ;NO - MOVE POINTER R1
000752                          BR      LOP5AA                  ;CHECK TO SEE IF ANY OTHER ADDRESS
                                                                ;ALSO RESPONDS TO THE ADDRESS POINTED
                                                                ;TO BY R1
000100  013652          DONE5A: MOV     #100,ICOUNT             ;DROP ITERATION COUNT

                                ;SHOW THAT BYTE ADDRESSING OF PAR'S WORKS FOR HIGH AND LOW BYTES
                        TEST6:  SCOPE
001000                          MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
014366                          JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
000006                          6                               ;TEST NUMBER
104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
002000  013626                  MOV     #2000,ICOUNT            ;RESTORE ITERATION COUNT
013334                          JSR     %7,CLRALL               ;INITIALIZE MEMORY MANAG. REGISTERS
001140                          MOV     #PARTAB,R3              ;R3 POINTS TO TABLE OF PAR ADDRESSES
```

```
                                                ;RO IS COUNTER OF STATES LEFT TO TEST
LOOP6:  MOV     (R3)+,R1                        ;PUT ADDRESS OF 1ST PAR IN SET IN R1
        MOV     #10,R2                          ;R2 IS COUNTER OF PAR'S LEFT TO TEST IN SET
LOOP6A: MOV     #-1,@R1                         ;SET UP PAR BEING TESTED
        CLRB    @R1                             ;CLEAR LOW BYTE OF PAR
        CMP     #7400,@R1                       ;CHECK PAR
        BEQ     .+4                             ;BRANCH IF OK
        HLT                                     ;DATOB TO PAR WHOSE ADDRESS IS IN
                                                ;R1 FAILED
        MOV     #-1,@R1                         ;SET UP PAR TO TEST HIGH BYTE
        CLRB    1(R1)                           ;CLEAR HIGH BYTE
        CMP     #377,@R1                        ;CHECK PAR
        BEQ     .+4
        HLT                                     ;DATOB TO HIGH BYTE OF PAR WHOSE
                                                ;ADDRESS IS IN R1 FAILED
        TST     (R1)+                           ;MOVE POINTER
        SOB     R2,LOOP6A                       ;TEST ALL PAR'S IN SET
        SOB     R0,LOOP6                        ;TEST ALL 2 REGISTER SETS

;SHOW THAT BYTE ADDRESSING OF PDR'S WORKS FOR HIGH AND LOW BYTES
TEST7:  SCOPE
        MOV     #KSTACK,SP                      ;INITIALIZE KERNEL STACK POINTER
        JSR     PC,ORDER                        ;CHECK TEST SEQUENCE + INIT SR0
        7                                       ;TEST NUMBER
        HLT                                     ;TEST EXECUTED OUT OF SEQUENCE
        JSR     %7,CLRALL                       ;INITIALIZE MEMORY MANAG. REGISTERS
        MOV     #PORTAB,R3                      ;R3 POINTS TO TABLE OF PDR ADDRESSES
        MOV     #2,R0                           ;R0 IS COUNTER OF STATES LEFT TO TEST
LOOP7:  MOV     (R3)+,R1                        ;PUT ADDRESS OF 1ST PDR IN SET INTO R1
        MOV     #10,R2                          ;R2 IS COUNTER OF PDR'S LEFT TO TEST IN SET
LOOP7A: MOV     #-1,@R1                         ;SET UP PDR BEING TESTED
        CLRB    @R1                             ;CLEAR LOW BYTE OF PDR
        CMP     #77400,@R1                      ;CHECK PDR
        BEQ     .+4                             ;BRANCH IF OK
        HLT                                     ;DATOB TO PDR WHOSE ADDRESS IS
                                                ;IN R1 FAILED
        MOV     #-1,@R1                         ;SET UP PDR TO TEST HIGH BYTE
        CLRB    1(R1)                           ;CLEAR HIGH BYTE
        CMP     #16,@R1                         ;CHECK PDR
        BEQ     .+4
        HLT                                     ;DATOB TO HIGH BYTE OF PDR WHOSE
                                                ;ADDRESS IS IN R1 FAILED
        TST     (R1)+                           ;MOVE POINTER
        SOB     R2,LOOP7A                       ;TEST ALL PDR'S IN SET
        SOB     R0,LOOP7                        ;TEST ALL 2 REGISTER SETS

;INIT SHOULD HAVE NO EFFECT ON PAR'S
TEST10: SCOPE
        MOV     #KSTACK,SP                      ;INITIALIZE KERNEL STACK POINTER
        JSR     PC,ORDER                        ;CHECK TEST SEQUENCE + INIT SR0
        10                                      ;TEST NUMBER
        HLT                                     ;TEST EXECUTED OUT OF SEQUENCE
        MOV     #10,ICOUNT                      ;DROP ITERATION COUNT
        CLR     TST10F
        MOV     #5252,R4
TST10:  MOV     #PARTAB,R3
```

```
000002  012700  000002          MOV     #2,R0
002266  012301                  LOOP10: MOV     (R3)+,R1
002270  012702  000010                  MOV     #10,R2          ;COUNTER TO LOAD PAR'S
002274  010421                  LOP10A: MOV     R4,(R1)+        ;LOAD PAR WITH PATTERN
002276  077202                          SOB     R2,LOP10A       ;LOAD ALL 16 IN THIS SET
002300  077006                          SOB     R0,LOOP10       ;INITIALIZE ALL 2 SETS
002302  000005                          RESET                   ;ISSUE INIT
002304  012703  001140                  MOV     #PARTAB,R3
002310  012700  000002                  MOV     #2,R0
002314  012301                  LOP10B: MOV     (R3)+,R1
002316  012702  000010                  MOV     #10,R2          ;COUNTER TO CHECK PAR'S
002322  020411                  LOP10C: CMP     R4,@R1          ;CHECK DATA
002324  001401                          BEQ     .+4
002326  104006                          HLT                     ;PAR WHOSE ADDRESS IS IN R1
                                                                ;WAS INCORRECT AFTER INIT
002330  005721                          TST     (R1)+           ;MOVE POINTER
002332  077205                          SOB     R2,LOP10C       ;TEST ALL 8 PAR'S IN THIS SET
002334  077011                          SOB     R0,LOP10B       ;TEST ALL 2 REGISTER SETS
002336  005767  000014                  TST     TST10F          ;CHECK FOR BOTH PATTERNS USED
002342  001006                          BNE     EXIT10          ;IF DONE, GO TO NEXT TEST
002344  005267  000006                  INC     TST10F          ;IF NOT, SET FLAG
002350  012704  002525                  MOV     #2525,R4        ;LOAD OTHER PATTERN
002354  000740                          BR      TST10           ;REPEAT TEST WITH 2ND PATTERN
002356  000000          TST10F: 0
002360                  EXIT10:


                        ;INIT SHOULDN'T CLEAR OR SET ANY OF THE R/W BITS IN THE PDR'S
002360  104400          TEST11: SCOPE
002362  012706  001000          MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
002366  004767  014012          JSR     PC,ORDER        ;CHECK TEST SEQUENCE + INIT SR0
002372  000011                  11                      ;TEST NUMBER
002374  104006                  HLT                     ;TEST EXECUTED OUT OF SEQUENCE
002376  005067  000104          CLR     TST11F
002402  012704  025012          MOV     #25012,R4       ;LOAD PATTERN IN R4
002406  012703  001134  TST11:  MOV     #PORTAB,R3
002412  012700  000002          MOV     #2,R0
002416  012301          LOOP11: MOV     (R3)+,R1
002420  012702  000010          MOV     #10,R2          ;COUNTER TO LOAD PDR'S
002424  010421          LOP11A: MOV     R4,(R1)+        ;LOAD PDR WITH PATTERN
002426  077202                  SOB     R2,LOP11A       ;LOAD ALL 8 IN THIS SET
002430  077006                  SOB     R0,LOOP11       ;INITIALIZE ALL 2 SETS
002432  000005                  RESET                   ;ISSUE INIT
002434  012703  001134          MOV     #PORTAB,R3
002440  012700  000002          MOV     #2,R0
002444  012301          LOP11B: MOV     (R3)+,R1
002446  012702  000010          MOV     #10,R2          ;COUNTER TO CHECK PDR'S
002452  020411          LOP11C: CMP     R4,@R1          ;CHECK DATA
002454  001401                  BEQ     .+4
002456  104006                  HLT                     ;PDR WHOSE ADDRESS IS IN R1
                                                        ;WAS INCORRECT AFTER INIT
002460  005721                  TST     (R1)+           ;MOVE POINTER
002462  077205                  SOB     R2,LOP11C       ;TEST ALL 8 PDR'S IN THIS SET
002464  077011                  SOB     R0,LOP11B       ;TEST ALL 2 REGISTER SETS
002466  005767  000014          TST     TST11F          ;CHECK FOR BOTH PATTERNS USED
002472  001006                  BNE     EXIT11          ;IF DONE, GO TO NEXT TEST
002474  005267  000006          INC     TST11F          ;IF NOT, SET FLAG
```

```
                                                MOV     #52404,R4               ;LOAD 2ND PATTERN
                                                BR      TST11
                                        TST11F: 0
                                        EXIT11: NOP

                                        ;SHOW THAT SR1 IS ONLY = 0 AND CANNOT BE LOADED
                                        TEST12: SCOPE
                                                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
                                                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
                                                12                              ;TEST NUMBER
                                                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
                                                MOV     #2000,ICOUNT            ;RESTORE ITERATION COUNT
                                                MOV     #-1,@SR1                ;TRY TO LOAD SR1
                                                TST     @SR1
                                                BEQ     .+4
                                                HLT                             ;SR1 INCORRECT - SHOULD HAVE TRACKED
                                        ;SR2 SHOULD CONTAIN ADDRESS OF LAST FETCH WITH MEMORY MANAG. TURNED OFF
                                        ;CHECK THAT ABORT FREEZES SR2
                                        TEST13: SCOPE
                                                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
                                                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
                                                13                              ;TEST NUMBER
                                                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
                                        AD13:   MOV     @SR2,R1                 ;PICK UP SR2 - SHOULD CONTAIN ADDRESS
                                                CMP     #AD13,R1                ;OF THIS INSTRUCTION
                                                BEQ     .+4
                                                HLT                             ;SR2 DID NOT CONTAIN FETCH ADDRESS
                                        AD13A:  BIS     #BIT15,@SR0             ;SET NR ABORT
                                                NOP
                                                CMP     #AD13A,@SR2             ;CHECK IF SR2 FROZE
                                                BEQ     .+4
                                                HLT                             ;SR2 NOT BEING DISABLED BY NR ABORT
                                                BIC     #BIT15,@SR0             ;CLEAR NR ABORT
                                        AD13B:  BIS     #BIT14,@SR0             ;SET PL ABORT
                                                NOP
                                                CMP     #AD13B,@SR2             ;DID SR2 FREEZE
                                                BEQ     .+4
                                                HLT                             ;SR2 NOT BEING DISABLED BY PL ABORT
                                                BIC     #BIT14,@SR0             ;CLEAR PL ABORT
                                        AD13C:  BIS     #BIT13,@SR0             ;SET RO ABORT
                                                NOP
                                                CMP     #AD13C,@SR2             ;DID SR2 FREEZE
                                                BEQ     .+4
                                                HLT                             ;SR2 NOT BEING DISABLED BY RO ABORT

                                        ;SHOW THAT DESTINATION ONLY RELOCATION DOESN'T RELOCATE AN INSTRUCTION
                                        ;FETCH (ONE CASE), AND THAT RESET CLEARS SR0<8>
                                        ;AND TURNS OFF DESTINATION ONLY RELOCATION
                                        ;IF THAT MUCH WORKS, YOU'LL GET THRU TO THE NEXT TEST
                                        TEST14: SCOPE
                                                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
                                                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
                                                14                              ;TEST NUMBER
                                                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
                                                JSR     %7,CLRALL               ;THIS TEST SHOULDN'T GO THRU ANY PAR/PDR PAIR'S
                                                                                ;SO MAKE THEM ALL GIVE NON-RESIDENT
```

```
                                                                ;AND PAGE LENGTH ERRORS IF ACCESSED
002732  012777  001006  176134          MOV     #1006,@KPDR0  ;3 BLOCKS OF KERNEL PDR0 MUST BE MAPPED
                                                                ;TO ALLOW TRAPS AND ABORTS
002740  012767  000010  012710          MOV     #10,ICOUNT    ;DROP THE ITERATION COUNT
002746  012777  000400  176050          MOV     #400,@SR0     ;TURN ON DESTINATION ONLY RELOCATION
002754  000005                          RESET                 ;SHOULD CLEAR DEST ONLY  BIT, AND A
                                                                ;SOLID PLACE TO START
                                                                ;IF THE FETCH IS RELOCATED
                                                                ;THIS WILL GIVE A PL ABORT
002756  032777  000400  176040          BIT     #400,@SR0     ;IF MEMORY MANAG. STILL ON, THIS SHOULD CAUSE
002764  001401                          BEQ     .+4           ;PL AND NR ERRORS
002766  000000                          HALT                  ;IF MEMORY MANAG. IS OFF, BIT 8 OF SR0 READS
                                                                ;AS STILL SET OR ANOTHER BIT IS INCORRECT
                                                                ;IF MEMORY MANAG. IS ON, NO NR OR SL ABORT
                                                                ;OCCURRED AND RESET FAILED TO TURN MEMORY MANAG.
002770  005077  176030                  CLR     @SR0

                        ;SHOW THAT DESTINATION ONLY RELOCATION DOESN'T RELOCATE THE SOURCE
                        ;ADDRESS AND DOES RELOCATE THE DESTINATION
002774  104400          TEST15: SCOPE
002776  012706  001000          MOV     #KSTACK,SP    ;INITIALIZE KERNEL STACK POINTER
003002  004767  013376          JSR     PC,ORDER      ;CHECK TEST SEQUENCE + INIT SR0
003006  000015                  15                    ;TEST NUMBER
003010  104006                  HLT                   ;TEST EXECUTED OUT OF SEQUENCE
003012  012767  000010  012636  MOV     #10,ICOUNT    ;KEEP THE NUMBER OF LOOPS DOWN
003020  004767  012344          JSR     %7,CLRALL
003024  012777  000001  176062  MOV     #1,@KPAR0     ;OFFSET KERNEL PAR/PDR PAIR 0 ONE BLOCK FROM BAN
003032  012777  077406  176034  MOV     #77406,@KPDR0
003040  012701  003112          MOV     #DATA16,R1    ;LOAD A BANK 0 ADDRESS
003044  012777  000400  175752  MOV     #400,@SR0     ;TURN ON DESTINATION ONLY RELOCATION
003052  021111                  CMP     @R1,@R1       ;THIS TEST WILL FAIL IF BOTH ARE
003054  001001                  BNE     .+4           ;RELOCATED OR BOTH ARE NOT RELOCATED
003056  000000                  HALT                  ;SOURCE AND DESTINATION BOTH ADDRESSED SAME LOCA
003060  000005                  RESET                 ;TURN OFF DESTINATION-ONLY RELOCATION
003062  012701  003012          MOV     #DATA16-100,R1 ;LOAD DESTINATION ADDRESS MINUS RELOCATION FACTO
003066  012702  003112          MOV     #DATA16,R2    ;LOAD SOURCE ADDRESS
003072  012777  000400  175724  MOV     #400,@SR0     ;TURN ON DESTINATION-ONLY RELOCATION
003100  021211                  CMP     @R2,@R1       ;USE SAME INSTRUCTION AND ADDRESS
003102  001401                  BEQ     .+4           ;MODES AS BEFORE
003104  000000                  HALT                  ;DESTINATION NOT RELOCATED OR INCORRECTLY
                                                      ;RELOCATED OR SOURCE RELOCATED
003106  000005                  RESET                 ;TURN OFF RELOCATION
003110  000401                  BR      .+4
003112  132465          DATA16: 132465

                        ;SHOW THAT A DATO OF 0 TO BIT 8, SR0 THRU KERNEL PAGE 7 WILL
                        ;CLEAR THE DESTINATION ONLY RELOCATION BIT AND TURN OFF DESTINATION ONLY RELOCATION
003114  104400          TEST16: SCOPE
003116  012706  001000          MOV     #KSTACK,SP    ;INITIALIZE KERNEL STACK POINTER
003122  004767  013256          JSR     PC,ORDER      ;CHECK TEST SEQUENCE + INIT SR0
003126  000016                  16                    ;TEST NUMBER
003130  104006                  HLT                   ;TEST EXECUTED OUT OF SEQUENCE
003132  004767  012232          JSR     %7,CLRALL     ;INITIALIZE
003136  012777  000001  175750  MOV     #1,@KPAR0     ;MAP KERNEL PAR/PDR PAIR 0
                                                      ;TO BANK 0 OFFSET BY 1 PAGE
003144  012777  077406  175722  MOV     #77406,@KPDR0 ;USED TO PROVE MEMORY MANAG. IS
                                                      ;TURNED OFF AFTER CLEARING BIT 8, SR0
```

```
 809  003152  012701  003112                MOV     #DATA16,R1              ;SETUP R1 TO REFERENCE KERNEL PAR/PDR PAIR 0
 810  003156  004767  013306                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
 811  003162  016702  175636                MOV     SR0,R2                  ;SETUP R2 TO ADDRESS SR0
 812  003166  012777  000400  175630        MOV     #400,@SR0               ;TURN ON DESTINATION ONLY RELOCATION
 813  003174  005012                        CLR     @R2                     ;CLEAR SR0 THRU KERNEL PAR/PDR PAIR7
 814  003176  021111                        CMP     @R1,@R1                 ;SHOW THAT MEMORY MANAG. IS OFF
 815  003200  001401                        BEQ     .+4
 816  003202  000000                        HALT                            ;MEMORY MANAG. STILL ON
 817  003204  032777  000400  175612        BIT     #400,@SR0               ;SHOW THAT BIT 8, SR0 IS NOW ZERO
 818  003212  001402                        BEQ     .+6
 819  003214  104006                        HLT                             ;DESTINATION ONLY RELOCATION BIT IS STILL ON
 820  003216  000005                        RESET                           ;MAKE SURE THAT MEMORY MANAG. IS OFF
 821                                ;SHOW THAT A DATO OF 0 TO BIT 8, SR0 THRU USER PAGE 7
 822                                ;WILL TURN OFF DESTINATION - ONLY PAGING
 823  003220  004767  012144                JSR     %7,CLRALL               ;INITIALLY CLEAR ALL PAR/PDR PAIRS
 824  003224  012777  000001  175622        MOV     #1,@UPAR0               ;MAP USER 0 TO
 825  003232  012777  077406  175574        MOV     #77406,@UPDR0           ;BANK 0 OFFSET BY 1 PAGE, RW
 826  003240  012701  003112                MOV     #DATA16,R1              ;SETUP R1 TO REFERENCE USER 0
 827  003244  012777  007600  175620        MOV     #7600,@UPAR7            ;MAP USER 7 TO THE
 828  003252  012777  077406  175572        MOV     #77406,@UPDR7           ;EXTERNAL BANK
 829  003260  016702  175540                MOV     SR0,R2                  ;SETUP R2 TO ADDRESS SR0
 830  003264  012737  140000  177776        MOV     #140000,@#PS            ;SET MODE TO USER
 831  003272  012777  000400  175524        MOV     #400,@SR0               ;TURN ON DESTINATION - ONLY PAGING
 832  003300  005012                        CLR     @R2                     ;CLEAR SR0 THRU USER ASR7
 833  003302  021111                        CMP     @R1,@R1                 ;SHOW THAT MEMORY MANAG. IS OFF
 834  003304  001401                        BEQ     .+4
 835  003306  000777                        BR      .                       ;RELOCATION STILL ON
 836
 837                                ;SHOW THAT ALL PAGE BOUNDARY REFERENCES REFERENCE THE CORRECT PAR
 838                                ;AND RELOCATE CORRECTLY
 839                                ;USE DESTINATION - ONLY PAGING
 840                                ;MAP ALL PAR/PDR PAIR'S RESIDENT READ WRITE
 841
 842                                ;       R0      - POINTS TO THE ADDRESS OF THE CURRENT PAR IN THE ADDRESS TABLE
 843                                ;       R1      - CONTAINS VIRTUAL ADDRESS BEING USED TO REFERENCE START OF PAGE
 844                                ;       R2      - CONTAINS VIRTUAL ADDRESS BEING USED TO REFERENCE END OF PAGE
 845                                ;       R3
 846                                ;       R4
 847                                ;       R5      - USED TO REFERENCE SR0 TO TURN OFF DESTINATION ONLY PAGING
 848  003310  104400                TEST17: SCOPE
 849  003312  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
 850  003316  004767  013062                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
 851  003322  000017                        17                              ;TEST NUMBER
 852  003324  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
 853  003326  004767  012036                JSR     %7,CLRALL               ;INITIALIZE
 854  003332  004767  012056                JSR     %7,RWALL                ;MAKE ALL  PAR/PDR PAIR'S RW, BANK 0,4K
 855  003336  013767  017700  175614        MOV     @#17700,SAVEA           ;SAVE CONTENTS OF LOCATIONS TO BE USED
 856  003344  013767  017776  175610        MOV     @#17776,SAVEB
 857  003352  012737  123456  017700        MOV     #123456,@#17700         ;SET UP LOCATIONS TO BE REFERENCED
 858  003360  012737  134567  017776        MOV     #134567,@#17776
 859  003366  012703  001012                MOV     #K123,R3
 860  003372  012704  001014                MOV     #K134,R4
 861  003376  012767  000100  012252        MOV     #100,ICOUNT             ;CHANGE ITERATION COUNT
 862  003404  012737  140000  177776        MOV     #140000,@#PS            ;CHANGE TO USER
 863  003412  012706  000400                MOV     #USTACK,SP              ;SET UP USER STACK POINTER
 864  003416  005037  177776                CLR     @#PS                    ;RETURN TO KERNEL
```

```
  865  003422  012767  001144  175524            MOV     #STATAB,STAPNT          ;SET UP TO REFERENCE STATE TABLE
  866  003430  017700  175520            STAT20: MOV     @STAPNT,R0              ;PICK UP ADDRESS OF START OF
  867  003434  062700  000020                    ADD     #20,R0                  ;ADDRESS TABLE FOR NEW STATE
  868  003440  062767  000002  175506           ADD     #2,STAPNT
  869  003446  017737  175502  177776           MOV     @STAPNT,@#PS            ;SET UP NEW STATE
  870  003454  062767  000002  175472           ADD     #2,STAPNT
  871  003462  012767  000010  175466           MOV     #8.,PAGES               ;SET UP COUNTER OF ASR'S LEFT TO TEST
  872  003470  012770  007600  000016           MOV     #7600,@16(R0)           ;SET UP SEGMENTED REFERENCE TO SR0,
  873  003476  016705  175322                   MOV     SR0,R5                  ;USED TO TURN DESTINATION - ONLY PAGING OFF
  874  003502  005001                           CLR     R1
  875  003504  012702  000076                   MOV     #76,R2
  876  003510  012767  000200  175460  PAG20:  MOV     #128.,BLOCKS            ;SET UP BLOCK COUNT
  877  003516  012770  000177  000000           MOV     #177,@(R0)              ;SET UP PAR
  878  003524  022767  000001  175424           CMP     #1,PAGES                ;IS THIS PAGE 7? (WAS USED
  879                                                                           ;FOR REFERENCE TO SR0)
  880  003532  001005                           BNE     BLK20                   ;IF NOT, BRANCH
  881  003534  012770  007600  177776           MOV     #7600,@-2(R0)           ;YES, SET UP PAGE 6 FOR REFERENCES TO SR0
  882  003542  042705  020000                   BIC     #20000,R5               ;CHANGE R5 TO POINT TO SR0 THRU PAR/PDR PAIR6
  883  003546  012777  000400  175250  BLK20:  MOV     #400,@SR0               ;TURN ON DESTINATION ONLY PAGING
  884  003554  021311                           CMP     @R3,@R1                 ;CK BOTTOM PAGE BOUNDARY
  885  003556  001401                           BEQ     .+4
  886  003560  000000                           HALT                            ;RELOCATION FAILED
  887  003562  021412                           CMP     @R4,@R2                 ;CK UPPER PAGE BOUNDARY
  888  003564  001401                           BEQ     .+4
  889  003566  000000                           HALT                            ;RELOCATION FAILED
  890  003570  005015                           CLR     @R5                     ;TURN OFF MEMORY MANAG.
  891  003572  005370  000000                   DEC     @(R0)                   ;MAP PAR 1 PAGE LOWER
  892  003576  062701  000100                   ADD     #100,R1                 ;SET UP R1 AND R2 TO REFERENCE
  893  003602  062702  000100                   ADD     #100,R2                 ;NEXT VIRTUAL PAGE
  894  003606  005367  175364                   DEC     BLOCKS                  ;DECREMENT COUNT OF PAGES LEFT
  895  003612  001355                           BNE     BLK20                   ;BRANCH IF NOT DONE WITH THIS PAR/PDR PAIR
  896  003614  005070  000000                   CLR     @(R0)
  897  003620  005367  175332                   DEC     PAGES                   ;DECREMENT COUNT OF PAR/PDR PAIR'S LEFT
  898  003624  001402                           BEQ     END20                   ;BRANCH IF ALL PAR/PDR PAIR'S IN THIS STATE DONE
  899  003626  005720                           TST     (R0)+                   ;MOVE ADDRESS TABLE POINTER
  900  003630  000727                           BR      PAG20
  901  003632  026727  175316  001152  END20:  CMP     STAPNT,#STAEND          ;CHECK FOR ALL STATES TESTED
  902  003640  003673                           BLE     STAT20                  ;IF NOT, BRANCH
  903  003642  005037  177776                   CLR     @#PS                    ;IF DONE, REINTIALIZE
  904  003646  005077  175152                   CLR     @SR0
  905  003652  016727  175302  017700           MOV     SAVEA,#17700
  906  003660  016727  175276  017776           MOV     SAVEB,#17776
  907  003666  016777  175274  175270           MOV     KTSTA,@KTVEC
  908  003674  005077  175266                   CLR     @KTSTA
  909
  910                                   ;SHOW THAT THE INSTRUCTIONS USED IN THE NEXT TEST RELOCATE CORRECTLY IN
  911                                   ;DESTINATION ONLY RELOCATION
  912  003700  104400                  TEST20: SCOPE
  913  003702  012706  001000                   MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
  914  003706  004767  012472                   JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
  915  003712  000020                           20                              ;TEST NUMBER
  916  003714  104006                           HLT                             ;TEST EXECUTED OUT OF SEQUENCE
  917  003716  012767  002000  011732           MOV     #2000,ICOUNT            ;RESTORE ITERATION COUNT
  918  003724  004767  011440                   JSR     %7,CLRALL               ;CLEAR ALL MEMORY MANAG. REGISTERS
  919  003730  012777  000001  175156           MOV     #1,@KPAR0               ;OFFSET KERNEL I-SPACE PAGE 0
  920  003736  012777  077406  175130           MOV     #77406,@KPDR0           ;BY 1 BLOCK FROM BANK 0
```

```
 921  003744  004767  012520                 JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
 922  003750  016701  175050                 MOV     SR0,R1                  ;SETUP R1 TO REFERENCE SR0
 923  003754  016746  000054                 MOV     DST21A-100,-(SP)
 924  003760  016746  000052                 MOV     DST21B-100,-(SP)
 925  003764  016746  000050                 MOV     DST21C-100,-(SP)
 926  003770  005067  000140                 CLR     DST21A                  ;INITIALIZE LOCATIONS TO BE
 927  003774  012767  177777  000134         MOV     #-1,DST21B              ;WRITTEN INTO
 928  004002  012767  177777  000130         MOV     #-1,DST21C
 929  004010  012777  000400  175006         MOV     #400,@SR0               ;TURN ON DESTINATION - ONLY RELOCATION
 930  004016  022737  176543  003720         CMP     #176543,@#AD21A-100     ;COMPARE THE CONTENTS OF AD21A
 931          004020                         AD21A=.-4                       ;WITH ITSELF, RELOCATED THRU KERNEL 0
 932  004024  001401                         BEQ     .+4
 933  004026  104006                         HLT                             ;DESTINATION - ONLY RELOCATION FAILED
 934                                                                         ;TO RELOCATE ONLY THE LAST CALCULATION
 935                                                                         ;OF THE CMP INSTRUCTION
 936  004030  122737  165432  003732         CMPB    #165432,@#AD21B-100     ;COMPARE THE CONTENTS OF AD21B
 937          004032                         AD21B=.-4                       ;WITH ITSELF, RELOCATED THRU KERNEL 0
 938  004036  001401                         BEQ     .+4
 939  004040  104006                         HLT                             ;DESTINATION - ONLY RELOCATION
 940                                                                         ;FAILED TO RELOCATE ONLY THE FINAL
 941                                                                         ;CALCULATION OF THE CMPB INSTRUCTION
 942  004042  012737  077711  004034         MOV     #77711,@#DST21A-100     ;EXECUTE REMAINING INSTRUCTIONS
 943  004050  005077  000066                 CLR     @AD21C
 944  004054  105037  004040                 CLRB    @#DST21C-100
 945  004060  005011                         CLR     @R1                     ;TURN OFF MEMORY MANAG.
 946  004062  022767  077711  000044         CMP     #77711,DST21A           ;CHECK LOCATION ADDRESSED BY MOV
 947  004070  001401                         BEQ     .+4
 948  004072  104006                         HLT                             ;MOV INSTRUCTION FAILED TO RELOCATE
 949                                                                         ;ONLY THE FINAL ADDRESS CALCULATION
 950  004074  005767  000036                 TST     DST21B                  ;CHECK LOCATION ADDRESSED BY CLR
 951  004100  001401                         BEQ     .+4
 952  004102  104006                         HLT                             ;CLR INSTRUCTION FAILED TO RELOCATE
 953                                                                         ;CORRECTLY IN DESTINATION - ONLY RELOCATION
 954  004104  022767  177400  000026         CMP     #177400,DST21C          ;CHECK LOCATION ADDRESSED BY CLRB
 955  004112  001401                         BEQ     .+4
 956  004114  104006                         HLT                             ;CLRB INSTRUCTION FAILED TO RELOCATE
 957                                                                         ;CORRECTLY IN DESTINATION - ONLY RELOCATION
 958  004116  012667  177716                 MOV     (SP)+,DST21C-100        ;RESTORE LOCATIONS IN CASE OF ERROR
 959  004122  012667  177710                 MOV     (SP)+,DST21B-100
 960  004126  012667  177702                 MOV     (SP)+,DST21A-100
 961  004132  000404                         BR      EXIT21
 962  004134  000000              DST21A:    0
 963  004136  000000              DST21B:    0
 964  004140  000000              DST21C:    0
 965  004142  004036              AD21C:     DST21B-100
 966  004144  000240              EXIT21:    NOP
 967
 968                                         ;TEST OF RELOCATION ADDERS - CHECK CORRECT PROPAGATION OF CARRY, AND CORRECT
 969                                         ;OUTPUT FOR EACH POSSIBLE COMBINATION FOR EACH BIT POSITION
 970                                         ;USE DESTINATION - ONLY RELOCATION, KERNEL
 971                                         ;TEST BY USING THE NECESSARY VALUE IN KERNEL PAR 1, WITH THE SECOND
 972                                         ;NECESPARY VALUE BEING THE VIRTUAL ADDRESS REFERENCE TO KERNEL PAR 1
 973                                         ;CHECK THE RESULTING PHYSICAL ADDRESS BY READING THE CONTENTS OF THE LOCATION,
 974                                         ;AND WRITING INTO THE LOCATION
 975                                         ;NOTE THAT THIS INCLUDES CHECKS OF ADDRESS WRAP AROUND
 976  004146  104400              TEST21:    SCOPE
```

```
  977  004150  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
  978  004154  004767  012224                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
  979  004160  000021                         21                              ;TEST NUMBER
  980  004162  104006                         HLT                             ;TEST EXECUTED OUT OF SEQUENCE
  981  004164  004767  011200                JSR     %7,CLRALL               ;CLEAR ALL MEMORY MANAG. REGISTERS
  982  004170  012777  077406  174676        MOV     #77406,@KPDR0           ;MAP KERNEL 0 TO BANK 0, 4K, RW
  983  004176  012777  077406  174672        MOV     #77406,@KPDR1           ;MAKE KERNEL 1 4K, RW
  984  004204  004767  012260                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
  985
  986                                         ;CHECK VIRTUAL ADDRESS OF 0 ADDED TO PAR OF -1 (FOR BIT POSITIONS
  987                                         ;RELEVANT TO THE ADDERS ONLY)
  988  004210  012777  007777  174700        MOV     #7777,@KPAR1            ;SET PAR TO -1
  989  004216  012737  030000  177776        MOV     #30000,@#PS             ;SET UP LOCATION TO BE REFERENCED
  990  004224  012777  000400  174572        MOV     #400,@SR0               ;TURN ON DESTINATION - ONLY PAGING
  991  004232  122737  000060  020077        CMPB    #60,@#20077             ;CHECK HIGH BYTE OF RESULTING ADDRESS
  992                                                                         ;(PS) - REFERENCED THRU PAR/PDR PAIR 1
  993  004240  001011                         BNE     ERR22A                  ;BRANCH ON FAILURE
  994  004242  105037  020077                CLRB    @#20077                 ;CLEAR PA 777777 THRU KERNEL 1
  995  004246  005077  174552                CLR     @SR0                    ;TURN OFF MEMORY MANAG.
  996  004252  105737  177777                TSTB    @#PS+1                  ;CHECK TO SEE IF CORRECT LOCATION
  997  004256  001401                         BEQ     .+4                     ;WAS REFERENCED
  998  004260  104006                         HLT                             ;RELOCATION FAILED
  999  004262  000405                         BR      CNT22B                  ;GO TO NEXT CHECK
 1000  004264  005077  174534        ERR22A: CLR     @SR0                    ;TURN OFF MEMORY MANAG.
 1001  004270  104006                         HLT                             ;RELOCATION FAILED IN THE COMPARE
 1002                                                                         ;AT LOCATION ADR22A
 1003  004272  005037  177776                CLR     @#PS                    ;REINITIALIZE PROCESSOR STATUS
 1004
 1005                                         ;CHECK VIRTUAL ADDRESS OF -1 ADDED TO PAR OF 0 (VALUES FOR BIT
 1006                                         ;POSITIONS RELEVANT TO THE ADDERS ONLY). RESULT SHOULD BE PA 17712
 1007  004276  005077  174614        CNT22B: CLR     @KPAR1                  ;SET PAR TO 0
 1008  004302  012737  125252  017712        MOV     #125252,@#DESTAD        ;LOAD PHYSICAL LOCATION TO BE REFERENCED
 1009                                                                         ;ADDRESS 17712
 1010  004310  012777  000400  174506        MOV     #400,@SR0               ;TURN ON DESTINATION - ONLY PAGING
 1011  004316  022737  125252  037712        CMP     #125252,@#37712         ;RELOCATE THRU KERNEL PAR/PDR PAIR1
 1012  004324  001011                         BNE     ERR22B                  ;BRANCH ON FAILURE
 1013  004326  005037  037712                CLR     @#37712                 ;CLEAR THRU KERNEL PAR/PDR PAIR1
 1014  004332  005077  174466                CLR     @SR0                    ;TURN OFF MEMORY MANAG.
 1015  004336  005737  017712                TST     @#17712                 ;CHECK TO SEE IF CORRECT LOCATION
 1016  004342  001401                         BEQ     .+4                     ;WAS CLEARED
 1017  004344  104006                         HLT                             ;RELOCATION FAILED
 1018  004346  000403                         BR      CNT22C                  ;GO TO NEXT CHECK
 1019  004350  005077  174450        ERR22B: CLR     @SR0                    ;TURN OFF MEMORY MANAG.
 1020  004354  104006                         HLT                             ;RELOCATION FAILED IN THE COMPARE
 1021                                                                         ;AT LOCATION ADR22B
 1022
 1023                                         ;CHECK VIRTUAL ADDRESS OF 1 (BIT 6) ADDED TO PAR OF -1
 1024                                         ;RESULTING PHYSICAL ADDRESS SHOULD BE ZERO
 1025                                         ;NOTE THAT THIS IS A CHECK OF ADDRESS WRAP AROUND
 1026  004356  012777  007777  174532        MOV     #7777,@KPAR1            ;SET UP PAR TO -1
 1027  004364  012737  034343  000000        MOV     #34343,@#0              ;SET UP A VALUE IN LOCATION TO
 1028                                                                         ;BE REFERENCED (0)
 1029  004372  012777  000400  174424        MOV     #400,@SR0               ;TURN ON DESTINATION-ONLY PAGING
 1030  004400  022737  034343  020100        CMP     #34343,@#20100          ;EFFECTIVELY ADDS 1 TO PAR ADDRESS
 1031                                                                         ;TO GET PHYSICAL ADDRESS OF 0
 1032  004406  001013                         BNE     ERR22C                  ;BRANCH ON FAILURE
```

```
1033  004410  012737  000002  020100          MOV     #2,@#20100      ;WRITE SAME LOCATION
1034  004416  005077  174402                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1035  004422  022737  000002  000000          CMP     #2,@#0          ;CHECK LOCATION WHICH SHOULD HAVE
1036  004430  001401                          BEQ     .+4             ;BEEN REFERENCED
1037  004432  104006                          HLT                     ;RELOCATION FAILED WHEN WRITING PA 0
1038  004434  000406                          BR      CNT22D          ;GO TO NEXT CHECK
1039  004436  005077  174362          ERR22C: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1040  004442  104006                          HLT                     ;RELOCATION FAILED IN THE COMPARE
1041  004444  012737  000002  000000          MOV     #2,@#0          ;AT LOCATION ADR22C
1042
1043                                          ;CHECK VIRTUAL ADDRESS OF -1 (BITS 6-12) ADDED TO PAR OF 1
1044                                          ;(PLUS HIGH BITS SET, BUT THEY DON'T ALTER CARRY CONDITION TESTED FOR)
1045                                          ;RESULTING PHYSICAL ADDRESS SHOULD BE ZERO
1046  004452  012777  007601  174436  CNT22D: MOV     #7601,@KPAR1    ;SET UP PAR TO 1, WITH HIGH BITS SET
1047  004460  012737  043434  000000          MOV     #43434,@#0      ;SET UP A VALUE IN LOCATION TO
1048                                                                  ;BE REFERENCED (0)
1049  004466  012777  000400  174330          MOV     #400,@SR0       ;TURN ON DESTINATION-ONLY PAGING
1050  004474  022737  043434  037700          CMP     #43434,@#37700  ;ALL HIGH BITS OF VA ARE 1, ADDED TO
1051                                                                  ;A ONE IN LOWEST BIT OF PAR TO PROPAGATE
1052                                                                  ;CARRY - RESULTING PHYSICAL ADDRESS 0
1053  004502  001013                          BNE     ERR22D          ;BRANCH ON FAILURE
1054  004504  012737  000002  037700          MOV     #2,@#37700      ;WRITE SAME LOCATION
1055  004512  005077  174306                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1056  004516  022737  000002  000000          CMP     #2,@#0          ;CHECK LOCATION WHICH SHOULD HAVE
1057                                                                  ;BEEN REFERENCED
1058  004524  001401                          BEQ     .+4
1059  004526  104006                          HLT                     ;RELOCATION FAILED WHEN WRITING PA 0
1060  004530  000406                          BR      CNT22E          ;GO TO NEXT CHECK
1061  004532  005077  174266          ERR22D: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1062  004536  104006                          HLT                     ;RELOCATION FAILED IN THE COMPARE
1063                                                                  ;AT LOCATION ADR22D
1064  004540  012737  000002  000000          MOV     #2,@#0          ;RESTORE LOCATION REFERENCED
1065
1066                                          ;CHECK VIRTUAL ADDRESS -1 ADDED TO PAR OF -1
1067                                          ;SHOULD GIVE RESULTING PA 17677
1068                                          ;NOTE THAT THIS IS A CASE OF ADDRESS WRAP AROUND
1069  004546  012777  007777  174342  CNT22E: MOV     #7777,@KPAR1    ;SET UP PAR TO -1
1070  004554  013746  017676                  MOV     @#17676,-(SP)   ;SAVE CONTENTS OF LOCATION TO BE
1071                                                                  ;REFERENCED
1072  004560  012737  076767  017676          MOV     #76767,@#17676  ;LOAD LOCATION TO BE REFERENCED
1073  004566  012777  000400  174230          MOV     #400,@SR0       ;TURN ON DESTINATION-ONLY PAGING
1074  004574  122737  000175  037777          CMPB    #175,@#37777    ;READ LOCATION (VA=-1, PAR=-1)
1075                                                                  ;SHOULD GIVE PA 17677 (THRU KERNEL PAR1
1076  004602  001012                          BNE     ERR22E          ;BRANCH ON FAILURE
1077  004604  105037  037777                  CLRB    @#37777         ;WRITE SAME LOCATION
1078  004610  005077  174210                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1079  004614  022737  000367  017676          CMP     #367,@#17676    ;CHECK TO SEE IF CORRECT LOCATION
1080  004622  001401                          BEQ     .+4             ;WAS CLEARED (HIGH BYTE)
1081  004624  104006                          HLT                     ;RELOCATION FAILED WHEN CLEARING
1082                                                                  ;PHYSICAL ADDRESS 17667 (THRU
1083  004626  000403                          BR      END22E          ;KERNEL PAR1)
1084  004630  005077  174170          ERR22E: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1085  004634  104006                          HLT                     ;RELOCATION FAILED IN THE COMPARE AT
1086                                                                  ;LOCATION ADR22E
1087  004636  012637  017676          END22E: MOV     (SP)+,@#17676   ;RESTORE LOCATION REFERENCED
1088
```

```
1089                                    ;SHOW THAT SETTING SR0<0> TURNS ON FULL RELOCATION
1090                                    ;SHOW THAT ALL ADDRESS CALCULATIONS ARE RELOCATED
1091                                    ;SHOW THAT INIT CLEARS SR0<0> AND TURNS OFF RELOCATION
1092  004642  104400                    TEST22: SCOPE
1093  004644  012706  001000                    MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
1094  004650  004767  011530                    JSR     PC,ORDER        ;CHECK TEST SEQUENCE + INIT SR0
1095  004654  000022                            22                      ;TEST NUMBER
1096  004656  104006                            HLT                     ;TEST EXECUTED OUT OF SEQUENCE
1097  004660  012767  000010  010770            MOV     #10,ICOUNT      ;DROP ITERATION COUNT
1098  004666  004767  010476                    JSR     %7,CLRALL       ;INITIALLY CLEAR ALL MEMORY MANAG. REGISTERS
1099  004672  012777  000001  174214            MOV     #1,@KPAR0       ;MAP KERNEL PAGE 0 TO
1100  004700  012777  077406  174166            MOV     #77406,@KPDR0   ;BANK 0 OFFSET BY 1 BLOCK
1101  004706  004767  011556                    JSR     PC,KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1102  004712  012767  052525  012772            MOV     #52525,DESTAD   ;INITIALIZE LOCATION  TO BE REFERENCED
1103  004720  012777  000001  174076            MOV     #1,@SR0         ;TURN ON RELOCATION
1104  004726  000000                    ADD23:  HALT                    ;WITH RELOCATION ON, SHOULD FETCH
1105  004730  000000                            HALT                    ;FROM 1 BLOCK ABOVE THIS
1106  004732  000000                            HALT                    ;(ADD23A)
1107  004734  000000                            HALT
1108  004736  000000                            HALT
1109  004740  000000                            HALT
1110  004742  032777  000001  174054            BIT     #1,@SR0         ;WHEN MEMORY MANAG. IS TURNED OFF, NEXT
1111                                                                    ;FETCH SHOULD BE FROM HERE -
1112  004750  001401                            BEQ     .+4             ;CHECK BIT 0, SR0
1113  004752  104006                            HLT                     ;MEMORY MANAG. OFF BUT SR0<0> STILL SET
1114                                                                    ;AFTER AN INIT
1115  004754  005077  174044                    CLR     @SR0
1116  004760  000432                            BR      EXIT23
1117  005026                                    .=ADD23+100
1118  005026  022737  052525  017612            CMP     #52525,@#DESTAD-100     ;WHEN MEMORY MANAG. IS TURNED ON, NEXT
1119                                                                    ;INSTRUCTION EXECUTED SHOULD
1120                                                                    ;BE HERE - CK RELOCATION OF SOURCE
1121  005034  001401                            BEQ     .+4             ;AND DESTINATION CALCULATIONS
1122  005036  000000                            HALT                    ;RELOCATION FAILED IN A SOURCE OR
1123                                                                    ;DESTINATION ADDRESS CALCULATION
1124                                                                    ;WITH FULL RELOCATION ON (SR0<0>SET)
1125  005040  000005                            RESET                   ;ISSUE INIT TO TURN OFF MEMORY MANAG.
1126  005042  000000                            HALT                    ;INIT DIDN'T TURN OFF MEMORY MANAG.
1127  005044  000777                            BR      .
1128  005046  000240                    EXIT23: NOP
1129
1130                                    ;SHOW THAT A DATO OF 0 TO SR0<0> WILL CLEAR SR0<0> AND
1131                                    ;TURN OFF RELOCATION
1132  005050  104400                    TEST23: SCOPE
1133  005052  012706  001000                    MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
1134  005056  004767  011322                    JSR     PC,ORDER        ;CHECK TEST SEQUENCE + INIT SR0
1135  005062  000023                            23                      ;TEST NUMBER
1136  005064  104006                            HLT                     ;TEST EXECUTED OUT OF SEQUENCE
1137  005066  012767  002000  010562            MOV     #2000,ICOUNT    ;RESTORE ITERATION COUNT
1138  005074  004767  010270                    JSR     %7,CLRALL       ;INITIALLY CLEAR ALL MEMORY MANAG. REGISTERS
1139  005100  012777  000001  174006            MOV     #1,@KPAR0       ;MAP KERNEL PAGE 0 TO
1140  005106  012777  077406  173760            MOV     #77406,@KPDR0   ;BANK 0 OFFSET BY 1 BLOCK
1141  005114  004767  011350                    JSR     PC,KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1142  005120  012777  000001  173676            MOV     #1,@SR0         ;TURN ON MEMORY MANAG.
1143  005126  000000                    ADD24:  HALT                    ;WHEN MEMORY MANAG. IS TURNED ON, SHOULD
1144  005130  000000                            HALT                    ;FETCH FROM THIS ADDRESS PLUS
```

```
1145  005132  000000                          HALT                    ;ONE BLOCK (ADD24A)
1146  005134  000240                          NOP
1147  005136  000240                          NOP
1148  005140  032777  000001  173656          BIT     #1,@SR0         ;AFTER MEMORY MANAG. IS TURNED OFF, CHECK
1149  005146  001401                          BEQ     .+4             ;SR0<0>
1150  005150  104006                          HLT                     ;MEMORY MANAG. OFF BUT SR0<0> STILL
1151                                                                  ;SET AFTER A BIC #1,@SR0
1152  005152  000433                          BR      EXIT24
1153
1154          005226                          .=ADD24+100
1155  005226  042777  000001  173570          BIC     #1,@SR0         ;WHEN MEMORY MANAG. IS TURNED ON, SHOULD
1156                                                                  ;RELOCATE FETCH TO HERE - TURN
1157                                                                  ;OFF MEMORY MANAG. VIA BIC OF SR0<0>
1158  005234  000000                          HALT                    ;MEMORY MANAG. STILL RELOCATING AFTER
1159  005236  000005                          RESET                   ;BIC OF SR0<0>
1160  005240  000777                          BR      .
1161  005242  000240                  EXIT24: NOP
1162
1163                                  ;SHOW THAT A REFERENCE TO A NON-RESIDENT PAGE
1164                                  ;WILL ABORT TO THE MEMORY MANAG. ABORT VECTOR ADDRESS (250)
1165                                  ;WITH BIT 15 OF SR0 SET.  SR0 AND SR2 ARE CHECKED FOR
1166                                  ;THE CORRECT VALUES, AS ARE KPDR0 AND KPDR1
1167                                  ;SHOW THAT BIT 15 OF SR0 CAN BE CLEARED
1168                                  ;SHOW THAT SR2 IS READ ONLY
1169
1170  005244  104400                  TEST24: SCOPE
1171  005246  012706  001000                  MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
1172  005252  004767  011126                  JSR     PC,ORDER        ;CHECK TEST SEQUENCE + INIT SR0
1173  005256  000024                          24                      ;TEST NUMBER
1174  005260  104006                          HLT                     ;TEST EXECUTED OUT OF SEQUENCE
1175  005262  004767  010102                  JSR     %7,CLRALL       ;CLEAR ALL MEMORY MANAG. REGISTERS
1176  005266  012777  077406  173600          MOV     #77406,@KPDR0   ;MAP KERNEL 0 TO BANK 0, RW, 4K
1177  005274  004767  011170                  JSR     PC,KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1178  005300  012777  005334  173656          MOV     #INT25,@KTVEC   ;SETUP RETURN VECTOR
1179  005306  005077  173654                  CLR     @KTSTA
1180  005312  012704  020000                  MOV     #20000,R4       ;USE R4 TO REFERENCE NR KERNEL 1
1181  005316  005277  173502                  INC     @SR0            ;TURN ON MEMORY MANAG.
1182  005322  005724                  ADR25:  TST     (R4)+           ;REFERENCE NR KERNEL 1
1183  005324  000000                  ADR25A: HALT                    ;SHOULD HAVE ABORTED ALREADY
1184  005326  005077  173472                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1185  005332  000442                          BR      DON25
1186  005334  017701  173464          INT25:  MOV     @SR0,R1         ;SAVE CONTENTS OF SR0
1187  005340  005377  173460                  DEC     @SR0            ;TURN OFF MEMORY MANAG.
1188  005344  022701  100003                  CMP     #100003,R1      ;CHECK SAVED CONTENTS OF SR0
1189  005350  001401                          BEQ     .+4
1190  005352  104006                          HLT                     ;SR0 INCORRECT AFTER NR ABORT
1191                                                                  ;(SEE SAVED CONTENTS IN R1)
1192  005354  022777  005322  173450          CMP     #ADR25,@SR2     ;CK SR2
1193  005362  001401                          BEQ     .+4
1194  005364  104006                          HLT                     ;SR2 INCORRECT-SHOULD CONTAIN ADDRESS
1195                                                                  ;OF LAST FETCH BEFORE THE ABORT
1196  005366  005077  173440                  CLR     @SR2            ;TRY TO WRITE INTO SR2
1197  005372  022777  005322  173432          CMP     #ADR25,@SR2     ;SR2 SHOULD BE READ ONLY
1198  005400  001401                          BEQ     .+4
1199  005402  104006                          HLT                     ;SR2 NOT READ ONLY
1200  005404  022777  077506  173462          CMP     #77506,@KPDR0
```

```
1201  005412  001401                          BEQ     .+4
1202  005414  104006                          HLT                             ;KERNEL PDR 0 INCORRECT
1203                                                                          ;W BIT SHOULD HAVE BEEN SET BY THE STACK WRITE
1204  005416  005777  173454                  TST     @KPDR1
1205  005422  001401                          BEQ     .+4
1206  005424  104006                          HLT                             ;KERNEL PDR 1 INCORRECT
1207  005426  021627  005324                  CMP     (R6),#ADR25A            ;CHECK VALUE PUSHED ON STACK
1208  005432  001401                          BEQ     .+4
1209  005434  104006                          HLT                             ;INCORRECT VALUE ON STACK
1210  005436  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK
1211  005440  005077  173522          DON25:  CLR     @KTSTA                  ;CHANGE TRAP VECTOR TO CAUSE A
1212  005444  016777  173516  173512          MOV     KTSTA,@KTVEC            ;HALT ON A FALSE TRAP
1213                                  ;SHOW THAT WRITING A PAGE WILL SET THE W BIT IN THE CORRESPONDING
1214                                  ;PDR, AND THAT NO OTHER W BITS SET AT THE SAME TIME
1215                                  ;SHOW THAT WRITING THE PDR (VIA A DATO) WILL CLEAR THE W BIT
1216                                  ;SINCE THIS IS DONE FOR ALL PDR'S, THIS IS ALSO
1217                                  ;A TEST OF INDIRECT ADDRESSING (VIA A VIRTUAL ADDRESS) OF THE PDR'S
1218  005452  104400                  TEST25: SCOPE
1219  005454  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1220  005460  004767  010720                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1221  005464  000025                          25                              ;TEST NUMBER
1222  005466  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1223  005470  012767  000400  010160          MOV     #400,ICOUNT             ;LOAD ITERATION COUNT
1224  005476  004767  007712                  JSR     %7,RWALL                ;MAP ALL PAR/PDR PAIR'S 4K, BANK 0, RW
1225  005502  004767  010762                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1226  005506  012777  007600  173356          MOV     #7600,@UPAR7            ;MAP USER 7 TO EXTERNAL BANK
1227  005514  012737  140000  177776          MOV     #140000,@#PS            ;SET MODE TO USER
1228  005522  012706  000400                  MOV     #USTACK,R6              ;SET UP USER STACK
1229  005526  005037  177776                  CLR     @#PS                    ;REINITIALIZE STATUS TO KERNEL MODE
1230  005532  012704  001034                  MOV     #ADRTAB,R4
1231  005536  012705  000010          LOP31A: MOV     #10,R5                  ;LOAD R4 WITH ADDRESS OF ADR TABLE
1232  005542  022734  077406          LOP31B: CMP     #77406,@(R4)+           ;INIT COUNTER OF PDR'S LEFT TO CHECK
1233  005546  001401                          BEQ     .+4                     ;CHECK ALL PDR W BITS BITS CLEAR
1234  005550  104006                          HLT                             ;PDR INCORRECT - W BIT SET OR ANOTHER
1235                                                                          ;BIT INCORRECT IN PDR WHOSE ADDRESS
1236  005552  077505                          SOB     R5,LOP31B               ;IS IN THE LOCATION POINTED TO BY R4
1237  005554  062704  000020                  ADD     #20,R4                  ;MOVE POINTER TO FIRST ADR OF NEXT SET
1238  005560  020427  001132                  CMP     R4,#ADREND
1239  005564  003003                          BGT     CNT31A                  ;BRANCH IF DONE
1240  005566  000763                          BR      LOP31A
1241  005570  012700  001144          CNT31A: MOV     #STATAB,R0              ;SET UP START OF STATE TABLE
1242  005574  012001                  LOP31C: MOV     (R0)+,R1        ;R1 CONTAINS ADDRESS OF PDR OF ADDRESS
1243  005576  012702  017776                  MOV     #17776,R2               ;SET UP VIRTUAL ADDRESS TO BE REFERENCED
1244  005602  012037  177776                  MOV     (R0)+,@#PS              ;SET UP STATUS FOR CURRENT MODE
1245  005606  005277  173212          LOP31D: INC     @SR0            ;TURN ON MEMORY MANAG.
1246  005612  011212                          MOV     (R2),(R2)               ;REFERENCE PAGE TO SET W BIT
1247  005614  005077  173204                  CLR     @SR0                    ;TURN OFF MEMORY MANAG.
1248  005620  032771  000100  000000          BIT     #100,@(R1)              ;CHECK W BIT
1249  005626  001001                          BNE     .+4
1250  005630  104006                          HLT                             ;W BIT NOT SET IN PDR AFTER PAGE WRITTEN
1251  005632  012703  001034                  MOV     #ADRTAB,R3              ;SET UP ADDRESS OF ADDRESS TABLE
1252  005636  012704  000010          LOP31E: MOV     #10,R4                  ;NOW CHECK ALL PDR TO SHOW NO OTHER
1253  005642  020103                  LOP31F: CMP     R1,R3                   ;W BITS WERE SET
1254  005644  001405                          BEQ     CNT31B
1255  005646  032773  000100  000000          BIT     #100,@(R3)
1256  005654  001401                          BEQ     .+4
```

```
1257  005656  104006                          HLT                             ;W BIT SET IN THE PDR WHOSE ADDRESS IS POINTED T
1258                                                                          ;AS WELL AS THE W BIT IN THE PDR
1259                                                                          ;FOR THE PAGE THAT WAS WRITTEN
1260  005660  005723                CNT31B:  TST     (R3)+                   ;UPDATE ADDRESS POINTER
1261  005662  077411                         SOB     R4,LOP31F               ;TEST NEW PDW
1262  005664  062703  000020                ADD     #20,R3                  ;UPDATE POINTER TO NEXT SET
1263  005670  020327  001132                CMP     R3,#ADREND
1264  005674  002760                        BLT     LOP31E
1265  005676  012771  077406  000000       MOV     #77406,@(R1)            ;CLEAR W BIT VIA DATO TO PDR
1266  005704  032771  000100  000000       BIT     #100,@(R1)              ;CHECK W BIT
1267  005712  001401                        BEQ     .+4
1268  005714  104006                        HLT                             ;W BIT DIDN'T CLEAR WHEN PDR
1269                                                                          ;WAS WRITTEN (ADDRESS OF ADDRESS
1270                                                                          ;OF PDR IS IN R1)
1271  005716  005721                        TST     (R1)+                   ;UPDATE POINTER
1272  005720  062702  020000                ADD     #20000,R2               ;CHANGE VA TO REFERENCE NEXT PAGE
1273  005724  103330                        BCC     LOP31D                  ;BRANCH TO TEST NEXT PAGE IN THIS MODE
1274  005726  020027  001152                CMP     R0,#STAEND              ;IF DONE THIS MODE, CHECK NEXT MODE
1275  005732  002720                        BLT     LOP31C                  ;LOOP UNTIL ALL STATES HAVE BEEN TESTED
1276  005734  005077  173064                CLR     @SR0                    ;REINITIALIZE SR0
1277
1278                                 ;SHOW THAT A REFERENCE TO A NR PAGE WILL SET BOTH THE NR AND PL
1279                                 ;ERROR BITS IF IT IS OUTSIDE THE MAPPED PAGE LENGTH
1280  005740  104400                TEST26:  SCOPE
1281  005742  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1282  005746  004767  010432                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1283  005752  000026                        26                              ;TEST NUMBER
1284  005754  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1285  005756  004767  007432                JSR     %7,RWALL                ;MAP ALL PAGES RW,4K,BANK 0
1286  005762  012777  000004  173106       MOV     #4,@KPDR1               ;MAP KERNEL 1 NR, 1 PAGE
1287  005770  004767  010474                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1288  005774  012777  006020  173162       MOV     #RET33,@KTVEC           ;SETUP ABORT RETURN
1289  006002  005077  173160                CLR     @KTSTA
1290  006006  005277  173012                INC     @SR0                    ;TURN ON MEMORY MANAG.
1291  006012  005737  030000                TST     @#30000                 ;REFERENCE NR KERNEL 1 - SHOULD ABORT
1292  006016  000000                        HALT                            ;NO NR ABORT
1293  006020  022777  140003  172776 RET33:  CMP     #140003,@SR0            ;CHECK SR0
1294  006026  001401                        BEQ     .+4
1295  006030  104006                        HLT                             ;SR0 INCORRECT - SHOULD SHOW KERNEL
1296                                                                          ;PAGE 1, AND BOTH NR + PL ERRORS SET
1297  006032  005077  172766                CLR     @SR0
1298  006036  016777  173124  173120       MOV     KTSTA,@KTVEC            ;RESTORE TRAP CATCHER
1299
1300                                 ;SHOW THAT KERNEL AND USER STACKS ARE ACCESSED CORRECTLY. AN IOT IS DONE TO
1301                                 ;EACH MODE. THE LOCATION WRITTEN INTO WHEN THE STACK IS PUSHED
1302                                 ;SHOWS WHICH STACK WAS USED.
1303  006044  104400                TEST27:  SCOPE
1304  006046  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1305  006052  004767  010326                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1306  006056  000027                        27                              ;TEST NUMBER
1307  006060  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1308  006062  004767  007302                JSR     %7,CLRALL               ;INITIALIZE ALL MEMORY MANAG. REGISTERS
1309  006066  012706  000500                MOV     #500,SP                 ;SET THE KERNEL STACK TO VIRTUAL ADDRESS 500
1310  006072  012737  140000  177776       MOV     #140000,@#PS
1311  006100  012706  000100                MOV     #100,SP                 ;SET THE USER STACK TO VA 100
1312  006104  005037  177776                CLR     @#PS
```

```
 1313  006110  012777  077406  172756          MOV     #77406,@#KPDR0          ;MAP KERNEL, AND USER TO BANK 0,4K,RW
 1314  006116  012777  077406  172710          MOV     #77406,@#UPDR0
 1315  006124  012737  006172  000020          MOV     #KRET34,@#20            ;TEST USING IOT TRAP (THRU KERNEL SPACE)
 1316  006132  005037  000022                  CLR     @#22                    ;RETURN FROM TRAP IN KERNEL MODE
 1317  006136  016701  172662                  MOV     SR0,R1                  ;REFERENCE SR0 THRU R1
 1318  006142  004767  010322                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
 1319  006146  012777  077406  172676          MOV     #77406,@#UPDR7          ;MAP USER PAGE 7
 1320  006154  012777  007600  172710          MOV     #7600,@#UPAR7           ;TO THE EXTERNAL BANK
```

```
1321  006162  005277  172636          INC     @SR0            ;TURN ON MEMORY MANAG.
1322  006166  000004                  IOT                     ;SHOULD USE STACK IN KERNEL ADDRESS SPACE
1323  006170  000240                  NOP
1324  006172  005011          KRET34: CLR     @R1             ;TURN OFF MEMORY MANAG.
1325  006174  012737  006226  000020  MOV     #URET34,@#20    ;SETUP FOR IOT TO USER
1326  006202  012737  140000  000022  MOV     #140000,@#22
1327  006210  012737  140000  177776  MOV     #140000,@#PS
1328  006216  005277  172602          INC     @SR0            ;TURN ON MEMORY MANAG.
1329  006222  000004                  IOT                     ;SHOULD USE STACK IN USER SPACE
1330  006224  000240                  NOP
1331  006226  005011          URET34: CLR     @R1             ;TURN OFF MEMORY MANAG.
1332  006230  022737  006170  000474  CMP     #KRET34-2,@#474
1333  006236  001401                  BEQ     .+4
1334  006240  104006                  HLT                     ;KERNEL STACK CONTENTS WRONG. PC NOT WHERE IT
1335  006242  022737  000000  000476  CMP     #0,@#476        ;SHOULD HAVE BEEN PUSHED OR
```

```
1336  006250  001401                        BEQ     .+4                     ;VALUE WRONG
1337  006252  104006                        HLT                             ;KERNEL STACK WRONG-TRAP STATUS NOT
1338  006254  022737  006224  000074        CMP     #URET34-2,@#74          ;NOT WHERE IT SHOULD HAVE BEEN PUSHED
1339  006262  001401                        BEQ     .+4                     ;OR VALUE WRONG
1340  006264  104006                        HLT                             ;USER STACK WRONG-PC NOT WHERE
1341  006266  022737  140000  000076        CMP     #140000,@#76            ;IT SHOULD HAVE BEEN PUSHED
1342  006274  001401                        BEQ     .+4                     ;OR VALUE WRONG
1343  006276  104006                        HLT                             ;USER STACK WRONG-TRAP STATUS
1344                                                                        ;NOT WHERE IT SHOULD HAVE BEEN
1345                                                                        ;PUSHED OR VALUE WRONG
1346  006300  012737  000076  000074        MOV     #76,@#74                ;REINITIALIZE LOCATIONS CHECKED
1347  006306  005037  000076                CLR     @#76
1348  006312  012737  000476  000474        MOV     #476,@#474
1349  006320  005037  000476                CLR     @#476
1350  006324  012706  001000                MOV     #KSTACK,SP
1351
1352                                  ;SHOW THAT TRAP,EMT, AND INTERRUPTS TAKE VECTORS FROM KERNEL
1353                                  ;IRREGARDLESS OF THE MODE AT THE TIME OF THE TRAP SEQUENCE
1354                                  ;ALSO SHOW THAT ODD-ADDRESS TRAP (AN "INTERNAL"
1355                                  ;TRAP) TAKES ITS VECTOR FROM KERNEL
1356                                  ;NOTE THAT IF DUAL ADDRESSING OCCURS, THE ERROR
1357                                  ;ADDRESS WILL BE USED (THE 0 OVERRIDES THE 1)
1358  006330  104400                  TEST30: SCOPE
1359  006332  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1360  006336  004767  010042                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1361  006342  000030                        30                              ;TEST NUMBER
1362  006344  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1363  006346  005077  172452                CLR     @SR0
1364  006352  004767  007036                JSR     %7,RMALL                ;MAP ALL PAR/PDR PAIR'S RW, 4K, BANK 0
1365  006356  012777  000001  172470        MOV     #1,@UPAR0               ;OFFSET USER 0 1 PAGE
1366  006364  004767  010100                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1367  006370  012777  007600  172474        MOV     #7600,@UPAR7            ;MAP USER 7 TO THE EXTERNAL BANK
1368  006376  016701  172422                MOV     SR0,R1                  ;SETUP R1 TO REFERENCE SR0
1369  006402  012737  140000  177776        MOV     #140000,@#PS            ;SETUP USER STACK
1370  006410  012706  000400                MOV     #USTACK,SP
1371  006414  005037  177776                CLR     @#PS
1372  006420  012706  001000                MOV     #KSTACK,SP              ;SETUP THE KERNEL STACK POINTER
1373  006424  012737  006564  000130        MOV     #NG358,@#130            ;SETUP FAILURE RETURN
1374  006432  012737  006602  000030        MOV     #OK358,@#30             ;SETUP SUCCESS RETURN
1375  006440  005037  000132                CLR     @#132
1376  006444  005037  000032                CLR     @#32
1377  006450  012737  140000  177776        MOV     #140000,@#PS            ;SET MODE TO USER
1378  006456  005277  172342                INC     @SR0                    ;TURN ON MEMORY MANAG.
1379  006462  000000                        0
1380  006464  000000                        0
1381  006466  000000                        0
1382  006470  000000                        0
1383  006472  000000                        0
1384  006474  000000                        0
1385  006476  000000                        0
1386  006500  000000                        0
1387  006502  000000                        0
1388  006504  000000                        0
1389  006506  000000                        0
1390  006510  000000                        0
1391  006512  000000                        0
```

```
1392  006514  000000                          0
1393  006516  000000                          0
1394  006520  000000                          0
1395  006522  000000                          0
1396  006524  000000                          0
1397  006526  000000                          0
1398  006530  000000                          0
1399  006532  000000                          0
1400  006534  000000                          0
1401  006536  000000                          0
1402  006540  000000                          0
1403  006542  000000                          0
1404  006544  000000                          0
1405  006546  000000                          0
1406  006550  000000                          0
1407  006552  000000                          0
1408  006554  000000                          0
1409  006556  000000                          0
1410  006560  000000                          0
1411  006562  104000                  EMT
1412  006564  022626              NG35B:  CMP     (SP)+,(SP)+             ;SHOULD PICK UP RETURN ADDRESS FROM KERNEL
1413  006566  005011                      CLR     @R1                     ;RESTORE STACK POINTER
1414  006570  012737  016334  000030      MOV     #EMTSRV,@#30            ;TURN OFF MEMORY MANAG.
1415  006576  104006                      HLT
1416  006600  000402                      BR      IOT35                   ;EMT VECTOR DIDN'T GO THRU KERNEL
1417  006602  022626              OK35B:  CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
1418  006604  005011                      CLR     @R1                     ;TURN OFF MEMORY MANAG.
1419  006606  012737  016334  000030  IOT35:  MOV     #EMTSRV,@#30            ;RESTORE EMT SERVICE POINTER
1420  006614  012737  000340  000032      MOV     #340,@#32
1421  006622  012737  000132  000130      MOV     #132,@#130
1422  006630  005037  000132              CLR     @#132
1423  006634  012737  006774  000120      MOV     #NG35C,@#120            ;SETUP IOT FAILURE RETURN
1424  006642  012737  007004  000020      MOV     #OK35C,@#20             ;SETUP IOT SUCCESS RETURN
1425  006650  005037  000122              CLR     @#122
1426  006654  005037  000022              CLR     @#22
1427  006660  012737  140000  177776      MOV     #140000,@#PS            ;SET MODE TO USER
1428  006666  005277  172132              INC     @SR0                    ;TURN ON MEMORY MANAG.
1429  006672  000000                          0
1430  006674  000000                          0
1431  006676  000000                          0
1432  006700  000000                          0
1433  006702  000000                          0
1434  006704  000000                          0
1435  006706  000000                          0
1436  006710  000000                          0
1437  006712  000000                          0
1438  006714  000000                          0
1439  006716  000000                          0
1440  006720  000000                          0
1441  006722  000000                          0
1442  006724  000000                          0
1443  006726  000000                          0
1444  006730  000000                          0
1445  006732  000000                          0
1446  006734  000000                          0
1447  006736  000000                          0
```

```
1448  006740  000000                          0
1449  006742  000000                          0
1450  006744  000000                          0
1451  006746  000000                          0
1452  006750  000000                          0
1453  006752  000000                          0
1454  006754  000000                          0
1455  006756  000000                          0
1456  006760  000000                          0
1457  006762  000000                          0
1458  006764  000000                          0
1459  006766  000000                          0
1460  006770  000000                          0
1461  006772  000004                          IOT                     ;SHOULD PICK UP RETURN ADDRESS FROM KERNEL
1462  006774  022626                  NG35C:  CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
1463  006776  005011                          CLR     @R1             ;TURN OFF MEMORY MANAG.
1464  007000  104006                          HLT                     ;TRAP VECTOR DIDN'T GO THRU KERNEL
1465  007002  000402                          BR      INT35
1466  007004  022626                  OK35C:  CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
1467  007006  005011                          CLR     @R1             ;TURN OFF MEMORY MANAG.
1468  007010  012737  000022  000020  INT35:  MOV     #22,@#20
1469  007016  005037  000022                  CLR     @#22
1470  007022  012737  000122  000120          MOV     #122,@#120
1471  007030  005037  000122                  CLR     @#122
1472  007034  012737  007216  000164          MOV     #NG35D,@#164    ;SETUP TTY FAILURE RETURN
1473  007042  012737  007232  000064          MOV     #OK35D,@#64     ;SETUP TTY SUCCESS RETURN
1474  007050  005037  000166                  CLR     @#166
1475  007054  005037  000066                  CLR     @#66
1476  007060  012737  140000  177776          MOV     #140000,@#PS    ;SET MODE TO USER
1477  007066  005277  171732                  INC     @SR0            ;TURN ON MEMORY MANAG.
1478  007072  000000                          0
1479  007074  000000                          0
1480  007076  000000                          0
1481  007100  000000                          0
1482  007102  000000                          0
1483  007104  000000                          0
1484  007106  000000                          0
1485  007110  000000                          0
1486  007112  000000                          0
1487  007114  000000                          0
1488  007116  000000                          0
1489  007120  000000                          0
1490  007122  000000                          0
1491  007124  000000                          0
1492  007126  000000                          0
1493  007130  000000                          0
1494  007132  000000                          0
1495  007134  000000                          0
1496  007136  000000                          0
1497  007140  000000                          0
1498  007142  000000                          0
1499  007144  000000                          0
1500  007146  000000                          0
1501  007150  000000                          0
1502  007152  000000                          0
1503  007154  000000                          0
```

```
1504  007156  000000                          0
1505  007160  000000                          0
1506  007162  000000                          0
1507  007164  000000                          0
1508  007166  000000                          0
1509  007170  000000                          0
1510  007172  012737  000100  177564          MOV     #100,@#177564           ;SET TTY INTERRUPT ENABLE-SHOULD
1511  007200  000240                          NOP                             ;INTERRUPT IMMEDIATELY
1512  007202  000240                          NOP
1513  007204  005011                          CLR     @R1                     ;TURN OFF MEMORY MANAG.
1514  007206  005077  171604                  CLR     @TCSR                   ;CLEAR TTY IE
1515  007212  104006                          HLT                             ;TTY FAILED TO INTERRUPT
1516  007214  000412                          BR      ODDAD
1517  007216  022626                  NG35D:  CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
1518  007220  005011                          CLR     @R1                     ;TURN OFF MEMORY MANAG.
1519  007222  005077  171570                  CLR     @TCSR                   ;CLEAR TTY IE
1520  007226  104006                          HLT                             ;TTY INTERRUPT DIDN'T GO THRU KERNEL
1521  007230  000404                          BR      ODDAD
1522  007232  022626                  OK35D:  CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
1523  007234  005011                          CLR     @R1                     ;TURN OFF MEMORY MANAG.
1524  007236  005077  171554                  CLR     @TCSR
1525  007242  012737  000066  000064  ODDAD:  MOV     #66,@#64                ;RESTORE TTY VECTOR RETURN TO CAUSE
1526  007250  005037  000066                  CLR     @#66                    ;A HALT ON A FALSE INTERRUPT
1527  007254  012737  000162  000160          MOV     #162,@#160
1528  007262  005037  000162                  CLR     @#162
1529  007266  005037  177776                  CLR     @#PS
1530  007272  012737  007436  000104          MOV     #NG35E,@#104            ;SETUP INTERNAL TRAP FAILURE RETURN
1531  007300  012737  000340  000106          MOV     #340,@#106
1532  007306  012737  007446  000004          MOV     #OK35E,@#4              ;SETUP INTERNAL TRAP SUCCESS RETURN
1533  007314  005037  000006                  CLR     @#6
1534  007320  012737  140000  177776          MOV     #140000,@#PS            ;SET MODE TO USER
1535  007326  005277  171472                  INC     @SR0                    ;TURN ON MEMORY MANAG.
1536  007332  000000                          0
1537  007334  000000                          0
1538  007336  000000                          0
1539  007340  000000                          0
1540  007342  000000                          0
1541  007344  000000                          0
1542  007346  000000                          0
1543  007350  000000                          0
1544  007352  000000                          0
1545  007354  000000                          0
1546  007356  000000                          0
1547  007360  000000                          0
1548  007362  000000                          0
1549  007364  000000                          0
1550  007366  000000                          0
1551  007370  000000                          0
1552  007372  000000                          0
1553  007374  000000                          0
1554  007376  000000                          0
1555  007400  000000                          0
1556  007402  000000                          0
1557  007404  000000                          0
1558  007406  000000                          0
1559  007410  000000                          0
```

```
1560  007412  000000                          0
1561  007414  000000                          0
1562  007416  000000                          0
1563  007420  000000                          0
1564  007422  000000                          0
1565  007424  000000                          0
1566  007426  000000                          0
1567  007430  000000                          0
1568  007432  005737  000001                  TST     @#1                     ;ODD ADDRESS REFERENCE - AN "INTERNAL
1569                                                                          ;TRAP" SHOULD OCCUR
1570  007436  022626                NG35E:    CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
1571  007440  005011                          CLR     @R1                     ;TURN OFF MEMORY MANAG.
1572  007442  104006                          HLT                             ;ODD ADDRESS TRAP DIDN'T TAKE
1573                                                                          ;VECTOR FROM KERNEL
1574  007444  000407                          BR      END35
1575  007446  022626                OK35E:    CMP     (SP)+,(SP)+             ;RESTORE STACK POINTER
1576  007450  005011                          CLR     @R1                     ;TURN OFF MEMORY MANAG.
1577  007452  032737  000340  177776          BIT     #340,@#PS               ;WAS CORRECT STATUS PICKED UP?
1578  007460  001401                          BEQ     .+4                     ;YES- BRANCH
1579  007462  104006                          HLT                             ;PICKED UP NEW STATUS WORD FROM USER SPACE
1580  007464  012737  000006  000004  END35:  MOV     #6,@#4                  ;RESTORE TRAP CATCHER
1581  007472  012737  000106  000104          MOV     #106,@#104
1582
1583
1584                                          ;SHOW THAT THE ABORT LOGIC "LOCKS"  SR0, AND SR2 AFTER A NR
1585                                          ;ABORT UNTIL THE CORRESPONDING ABORT BIT IS CLEARED IN SR0, WHEN
1586                                          ;THEY RESUME TRACKING. A NR ERROR SHOULD STILL ABORT TO 250 EVEN
1587                                          ;WHEN BIT 15 (SR0) IS ALREADY SET
1588  007500  104400                TEST31:   SCOPE
1589  007502  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1590  007506  004767  006672                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1591  007512  000031                          31                              ;TEST NUMBER
1592  007514  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1593  007516  004767  005646                  JSR     %7,CLRALL               ;CLEAR ALL MEMORY MANAG. REGISTERS
1594  007522  004767  006742                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1595  007526  012777  077406  171340          MOV     #77406,@KPDR0           ;MAP KERNEL 0 RW,RK,BANK0
1596  007534  012777  077400  171334          MOV     #77400,@KPDR1           ;MAP KERNEL 1 NR,4 K,BANK0
1597  007542  012777  007576  171414          MOV     #INT36,@KTVEC           ;SETUP RETURN VECTOR
1598  007550  005077  171412                  CLR     @KTSTA
1599  007554  005277  171244                  INC     @SR0                    ;TURN ON MEMORY MANAG.
1600  007560  013737  037776  037776  ADR36:  MOV     @#37776,@#37776         ;REFERENCE KERNEL 1 - 1ST ABORT
1601  007566  005077  171232                  CLR     @SR0                    ;TURN OFF MEMORY MANAG.
1602  007572  104006                          HLT                             ;REFERENCE TO KERNEL 1
1603  007574  000510                          BR      DONE36                  ;DIDN'T ABORT
1604  007576  042777  000001  171220  INT36:  BIC     #1,@SR0                 ;TURN OFF MEMORY MANAG.
1605  007604  022777  100002  171212          CMP     #100002,@SR0            ;CHECK SR0
1606  007612  001401                          BEQ     .+4
1607  007614  104006                          HLT                             ;SR0 INCORRECT AFTER NR ABORT
1608  007616  012777  007652  171340          MOV     #INT36A,@KTVEC          ;SETUP NEW RETURN VECTOR
1609  007624  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1610  007626  012702  037776                  MOV     #37776,R2               ;SETUP R2 TO REFERENCE KERNEL 1
1611  007632  052777  000001  171164          BIS     #1,@SR0                 ;TURN ON MEMORY MANAG.
1612  007640  012242                          MOV     (R2)+,-(R2)             ;REFERENCE KERNEL 1 -2ND ABORT
1613  007642  005077  171156          ADR36A: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1614  007646  104006                          HLT                             ;2ND REFERENCE TO KERNEL 1
1615  007650  000462                          BR      DONE36                  ;DIDN'T ABORT
```

```
1616  007652  042777  000001  171144  INT36A: BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1617  007660  022777  100002  171136          CMP     #100002,@SR0            ;CHECK SR0
1618  007666  001401                          BEQ     .+4
1619  007670  104006                          HLT                     ;SR0 INCORRECT AFTER 2ND NR ABORT
1620  007672  022777  007560  171132          CMP     #ADR36,@SR2             ;CHECK SR2
1621  007700  001401                          BEQ     .+4
1622  007702  104006                          HLT                     ;SR2 DOESN'T CONTAIN VALUE FROM 1ST ABORT
1623  007704  021627  007642                  CMP     (R6),#ADR36A            ;CHECK ADDRESS PUSHED ON STACK
1624  007710  001401                          BEQ     .+4
1625  007712  104006                          HLT                     ;INCORRECT ADDRESS ON STACK
1626  007714  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1627  007716  012777  007752  171240          MOV     #INT36B,@KTVEC          ;CHANGE RETURN ADDRESS
1628  007724  005077  171074                  CLR     @SR0                    ;CLEAR NR ERROR BIT-SHOULD
1629                                                                          ;"UNLOCK" ERROR TRACKING
1630  007730  012702  037776                  MOV     #37776,R2               ;SETUP R2 TO REFERENCE KERNEL 1
1631  007734  005277  171064                  INC     @SR0                    ;TURN ON MEMORY MANAG.
1632  007740  012242          ADR36B: MOV     (R2)+,-(R2)     ;3RD NR REFERENCE, ERROR BIT WAS CLEARED
1633  007742  005077  171056  ADR36C: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1634  007746  104006                          HLT                     ;3RD REFERENCE TO KERNEL 1
1635  007750  000422                          BR      DONE36          ;DIDN'T ABORT
1636  007752  042777  000001  171044  INT36B: BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1637  007760  022777  100002  171036          CMP     #100002,@SR0            ;CHECK SR0
1638  007766  001401                          BEQ     .+4
1639  007770  104006                          HLT                     ;SR0 INCORRECT
1640  007772  022777  007740  171032          CMP     #ADR36B,@SR2            ;CHECK SR2
1641  010000  001401                          BEQ     .+4
1642  010002  104006                          HLT                     ;SR2 INCORRECT - SHOULD CONTAIN
1643                                                                  ;LAST FETCH ADDRESS BEFORE ABORT
1644  010004  022716  007742                  CMP     #ADR36C,(SP)            ;CHECK STACK
1645  010010  001401                          BEQ     .+4
1646  010012  104006                          HLT                     ;PC ON STACK INCORRECT
1647  010014  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1648  010016  005077  171002          DONE36: CLR     @SR0                    ;CLEAR ERROR BIT
1649  010022  005077  171140                  CLR     @KTSTA                  ;CHANGE TRAP RETURN TO CAUSE A HALT
1650  010026  016777  171134  171130        MOV     KTSTA,@KTVEC            ;ON A FALSE INTERRUPT
1651
1652                          ;SHOW THAT THE ABORT LOGIC "LOCKS" SR0, AND SR2 AFTER A PL
1653                          ;ABORT UNTIL THE CORRESPONDING ABORT BIT IS CLEARED IN SR0, WHEN
1654                          ;THEY RESUME TRACKING. A PL ERROR SHOULD STILL ABORT TO 250 EVEN
1655                          ;WHEN BIT 14 (SR0) IS ALREADY SET
1656  010034  104400          TEST32: SCOPE
1657  010036  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1658  010042  004767  006336                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1659  010046  000032                          32                              ;TEST NUMBER
1660  010050  104006                          HLT                     ;TEST EXECUTED OUT OF SEQUENCE
1661  010052  004767  005312                  JSR     %7,CLRALL               ;CLEAR ALL MEMORY MANAG. REGISTERS
1662  010056  004767  006406                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1663  010062  012777  077406  171004          MOV     #77406,@KPDR0           ;MAP KERNEL 0 RW,RK,BANK0
1664  010070  012777  017406  171000          MOV     #17406,@KPDR1           ;MAP KERNEL 1 PL:1 R,BANK0
1665  010076  012777  010132  171060          MOV     #INT37,@KTVEC           ;SETUP RETURN VECTOR
1666  010104  005077  171056                  CLR     @KTSTA
1667  010110  005277  170710                  INC     @SR0                    ;TURN ON MEMORY MANAG.
1668  010114  013737  037776  037776  ADR37:  MOV     @#37776,@#37776         ;REFERENCE KERNEL 1 - 1ST ABORT
1669  010122  005077  170676                  CLR     @SR0                    ;TURN OFF MEMORY MANAG.
1670  010126  104006                          HLT                             ;REFERENCE TO KERNEL 1
1671  010130  000510                          BR      DONE37                  ;DIDN'T ABORT
```

```
1672  010132  042777  000001  170664  INT37:  BIC     #1,@SR0                 ;TURN OFF MEMORY MANAG.
1673  010140  022777  040002  170656          CMP     #40002,@SR0             ;CHECK SR0
1674  010146  001401                          BEQ     .+4
1675  010150  104006                          HLT                             ;SR0 INCORRECT AFTER PL ABORT
1676  010152  012777  010206  171004          MOV     #INT37A,@KTVEC          ;SETUP NEW RETURN VECTOR
1677  010160  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1678  010162  012702  037776                  MOV     #37776,R2               ;SETUP R2 TO REFERENCE KERNEL 1
1679  010166  052777  000001  170630          BIS     #1,@SR0                 ;TURN ON MEMORY MANAG.
1680  010174  012242                          MOV     (R2)+,-(R2)             ;REFERENCE KERNEL 1 -2ND ABORT
1681  010176  005077  170622          ADR37A: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1682  010202  104006                          HLT                             ;2ND REFERENCE TO KERNEL 1
1683  010204  000462                          BR      DONE37                  ;DIDN'T ABORT
1684  010206  042777  000001  170610  INT37A: BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1685  010214  022777  040002  170602          CMP     #40002,@SR0             ;CHECK SR0
1686  010222  001401                          BEQ     .+4
1687  010224  104006                          HLT                             ;SR0 INCORRECT AFTER 2ND PL ABORT
1688  010226  022777  010114  170576          CMP     #ADR37,@SR2             ;CHECK SR2
1689  010234  001401                          BEQ     .+4
1690  010236  104006                          HLT                             ;SR2 DOESN'T CONTAIN VALUE FROM 1ST ABORT
1691  010240  021627  010176                  CMP     (R6),#ADR37A            ;CHECK ADDRESS PUSHED ON STACK
1692  010244  001401                          BEQ     .+4
1693  010246  104006                          HLT                             ;INCORRECT ADDRESS ON STACK
1694  010250  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1695  010252  012777  010306  170704          MOV     #INT37B,@KTVEC          ;CHANGE RETURN ADDRESS
1696  010260  005077  170540                  CLR     @SR0                    ;CLEAR PL ERROR BIT-SHOULD
1697                                                                          ;"UNLOCK" ERROR TRACKING
1698  010264  012702  037776                  MOV     #37776,R2               ;SETUP R2 TO REFERENCE KERNEL 1
1699  010270  005277  170530                  INC     @SR0                    ;TURN ON MEMORY MANAG.
1700  010274  012242                  ADR37B: MOV     (R2)+,-(R2)     ;3RD PL REFERENCE, ERROR BIT WAS CLEARED
1701  010276  005077  170522          ADR37C: CLR     @SR0            ;TURN OFF MEMORY MANAG.
1702  010302  104006                          HLT                             ;3RD REFERENCE TO KERNEL 1
1703  010304  000422                          BR      DONE37                  ;DIDN'T ABORT
1704  010306  042777  000001  170510  INT37B: BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1705  010314  022777  040002  170502          CMP     #40002,@SR0             ;CHECK SR0
1706  010322  001401                          BEQ     .+4
1707  010324  104006                          HLT                             ;SR0 INCORRECT
1708  010326  022777  010274  170476          CMP     #ADR37B,@SR2            ;CHECK SR2
1709  010334  001401                          BEQ     .+4
1710  010336  104006                          HLT                             ;SR2 INCORRECT - SHOULD CONTAIN
1711                                                                          ;LAST FETCH ADDRESS BEFORE ABORT
1712  010340  022716  010276                  CMP     #ADR37C,(SP)            ;CHECK STACK
1713  010344  001401                          BEQ     .+4
1714  010346  104006                          HLT                             ;PC ON STACK INCORRECT
1715  010350  022626                          CMP     (R6)+,(R6)+             ;RESTORE STACK POINTER
1716  010352  005077  170446          DONE37: CLR     @SR0                    ;CLEAR ERROR BIT
1717  010356  005077  170604                  CLR     @KTSTA                  ;CHANGE TRAP RETURN TO CAUSE A HALT
1718  010362  016777  170600  170574          MOV     KTSTA,@KTVEC            ;ON A FALSE INTERRUPT
1719
1720                          ;SHOW THAT THE ABORT LOGIC "LOCKS"  SR0, AND SR2 AFTER A ACC
1721                          ;ABORT UNTIL THE CORRESPONDING ABORT BIT IS CLEARED IN SR0, WHEN
1722                          ;THEY RESUME TRACKING. A ACC ERROR SHOULD STILL ABORT TO 250 EVEN
1723                          ;WHEN BIT 13 (SR0) IS ALREADY SET
1724  010370  104400          TEST33: SCOPE
1725  010372  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1726  010376  004767  006002                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1727  010402  000033                          33                              ;TEST NUMBER
```

```
1728  010404  104006                        HLT     ;TEST EXECUTED OUT OF SEQUENCE
1729  010406  004767  004756                JSR     X7,CLRALL       ;CLEAR ALL MEMORY MANAG. REGISTERS
1730  010412  004767  006052                JSR     PC,KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1731  010416  012777  077406  170450        MOV     #77406,@KPDR0   ;MAP KERNEL 0 RW,RK,BANK0
1732  010424  012777  077402  170444        MOV     #77402,@KPDR1   ;MAP KERNEL 1 ACC,4'K,BANK0
1733  010432  012777  010466  170524        MOV     #INT40,@KTVEC   ;SETUP RETURN VECTOR
1734  010440  005077  170522                CLR     @KTSTA
1735  010444  005277  170354                INC     @SR0            ;TURN ON MEMORY MANAG.
1736  010450  013737  037776  037776  ADR40:MOV     @#37776,@#37776 ;REFERENCE KERNEL 1 - 1ST ABORT
1737  010456  005077  170342                CLR     @SR0            ;TURN OFF MEMORY MANAG.
1738  010462  104006                        HLT                     ;REFERENCE TO KERNEL 1
1739  010464  000510                        BR      DONE40          ;DIDN'T ABORT
1740  010466  042777  000001  170330  INT40:BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1741  010474  022777  020002  170322        CMP     #20002,@SR0     ;CHECK SR0
1742  010502  001401                        BEQ     .+4
1743  010504  104006                        HLT                     ;SR0 INCORRECT AFTER ACC ABORT
1744  010506  012777  010542  170450        MOV     #INT40A,@KTVEC  ;SETUP NEW RETURN VECTOR
1745  010514  022626                        CMP     (R6)+,(R6)+     ;RESTORE STACK POINTER
1746  010516  012702  037776                MOV     #37776,R2       ;SETUP R2 TO REFERENCE KERNEL 1
1747  010522  052777  000001  170274        BIS     #1,@SR0         ;TURN ON MEMORY MANAG.
1748  010530  012242                        MOV     (R2)+,-(R2)     ;REFERENCE KERNEL 1 -2ND ABORT
1749  010532  005077  170266  ADR40A:CLR     @SR0            ;TURN OFF MEMORY MANAG.
1750  010536  104006                        HLT                     ;2ND REFERENCE TO KERNEL 1
1751  010540  000462                        BR      DONE40          ;DIDN'T ABORT
1752  010542  042777  000001  170254  INT40A:BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1753  010550  022777  020002  170246        CMP     #20002,@SR0     ;CHECK SR0
1754  010556  001401                        BEQ     .+4
1755  010560  104006                        HLT                     ;SR0 INCORRECT AFTER 2ND ACC ABORT
1756  010562  022777  010450  170242        CMP     #ADR40,@SR2     ;CHECK SR2
1757  010570  001401                        BEQ     .+4
1758  010572  104006                        HLT                     ;SR2 DOESN'T CONTAIN VALUE FROM 1ST ABORT
1759  010574  021627  010532                CMP     (R6),#ADR40A    ;CHECK ADDRESS PUSHED ON STACK
1760  010600  001401                        BEQ     .+4
1761  010602  104006                        HLT                     ;INCORRECT ADDRESS ON STACK
1762  010604  022626                        CMP     (R6)+,(R6)+     ;RESTORE STACK POINTER
1763  010606  012777  010642  170350        MOV     #INT40B,@KTVEC  ;CHANGE RETURN ADDRESS
1764  010614  005077  170204                CLR     @SR0            ;CLEAR ACC ERROR BIT-SHOULD
1765                                                                ;"UNLOCK" ERROR TRACKING
1766  010620  012702  037776                MOV     #37776,R2       ;SETUP R2 TO REFERENCE KERNEL 1
1767  010624  005277  170174                INC     @SR0            ;TURN ON MEMORY MANAG.
1768  010630  012242          ADR40B:MOV    (R2)+,-(R2)     ;3RD ACC REFERENCE, ERROR BIT WAS CLEARED
1769  010632  005077  170166  ADR40C:CLR     @SR0            ;TURN OFF MEMORY MANAG.
1770  010636  104006                        HLT                     ;3RD REFERENCE TO KERNEL 1
1771  010640  000422                        BR      DONE40          ;DIDN'T ABORT
1772  010642  042777  000001  170154  INT40B:BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1773  010650  022777  020002  170146        CMP     #20002,@SR0     ;CHECK SR0
1774  010656  001401                        BEQ     .+4
1775  010660  104006                        HLT                     ;SR0 INCORRECT
1776  010662  022777  010630  170142        CMP     #ADR40B,@SR2    ;CHECK SR2
1777  010670  001401                        BEQ     .+4
1778  010672  104006                        HLT                     ;SR2 INCORRECT - SHOULD CONTAIN
1779                                                                ;LAST FETCH ADDRESS BEFORE ABORT
1780  010674  022716  010632                CMP     #ADR40C,(SP)    ;CHECK STACK
1781  010700  001401                        BEQ     .+4
1782  010702  104006                        HLT                     ;PC ON STACK INCORRECT
1783  010704  022626                        CMP     (R6)+,(R6)+     ;RESTORE STACK POINTER
```

```
1784  010706  005077  170112          DONE40: CLR     @SR0                    ;CLEAR ERROR BIT
1785  010712  005077  170250                  CLR     @KTSTA                  ;CHANGE TRAP RETURN TO CAUSE A HALT
1786  010716  016777  170244  170240          MOV     KTSTA,@KTVEC            ;ON A FALSE INTERRUPT
1787
1788                                  ;SHOW THAT INIT CLEARS SR0<13-15>
1789  010724  104400                  TEST34: SCOPE
1790  010726  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1791  010732  004767  005446                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1792  010736  000034                          34                              ;TEST NUMBER
1793  010740  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1794  010742  112777  000340  170056          MOVB    #340,@SR0H      ;SET SR0 BITS 13-15
1795  010750  122777  000340  170050          CMPB    #340,@SR0H      ;MAKE SURE THEY SET CORRECTLY
1796  010756  001401                          BEQ     .+4
1797  010760  104006                          HLT                             ;SR0 INCORRECT (HIGH BYTE)
1798  010762  000005                          RESET                           ;ISSUE INIT
1799  010764  122777  000000  170034          CMPB    #0,@SR0H        ;CHECK SR0 HIGH BYTE
1800  010772  001401                          BEQ     .+4
1801  010774  104006                          HLT                             ;SR0 INCORRECT AFTER INIT
1802  010776  012767  000010  004652          MOV     #10,ICOUNT      ;DROP ITERATION COUNT
1803
1804                                  ;SHOW THAT INIT CLEARS SR0 AFTER ABORT
1805  011004  104400                  TEST35: SCOPE
1806  011006  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1807  011012  004767  005366                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1808  011016  000035                          35                              ;TEST NUMBER
1809  011020  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1810  011022  004767  004366                  JSR     %7,RWALL        ;MAP ALL PAR/PDR PAIR'S 4K,RW,BANK 0
1811  011026  012777  000416  170040          MOV     #416,@KPDR0     ;MAP KERNEL 0 RW,4K LESS 1 PAGE
1812                                                                  ;DOWN (100-17776 RW)
1813  011034  004767  005430                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1814  011040  012777  077400  170030          MOV     #77400,@KPDR1   ;MAP KERNEL PAGE 1 NR
1815  011046  012777  011102  170110          MOV     #RET2,@KTVEC    ;SETUP ABORT RETURN
1816  011054  005077  170106                  CLR     @KTSTA
1817  011060  012746  000020                  MOV     #20,-(SP)       ;SET T BIT IN STATUS ON STACK
1818  011064  012746  011076                  MOV     #ADR2,-(SP)     ;SETUP ADDRESS ON STACK
1819  011070  005277  167730                  INC     @SR0            ;TURN ON MEMORY MANAG.
1820  011074  000002                          RTI                             ;SHOULD TRACE TRAP IMMEDIATELY SINCE T-BIT
1821                                                                  ;IS SET - SINCE T-BIT VECTOR IS OUTSIDE ALLOWED
1822                                                                  ;PAGE LENGTH, SHOULD DO A MEMORY
1823                                                                  ;MANAGEMENT ABORT
1824  011076  000000          ADR2:   HALT                            ;NO PL ABORT OCCURRED
1825  011100  000412                  BR      DONE2
1826  011102  022777  040001  167714  RET2:   CMP     #40001,@SR0     ;CHECK SR0
1827  011110  001401                          BEQ     .+4
1828  011112  104006                          HLT                             ;SR0 INCORRECT - SHOULD SHOW
1829                                                                  ;REFERENCE TO KERNEL 0
1830                                                                  ;AND PL ABORT SHOULD BE SET
1831  011114  000005                          RESET                           ;ISSUE INIT - SHOULD CLEAR SR0
1832  011116  005777  167702                  TST     @SR0            ;CHECK SR0
1833  011122  001401                          BEQ     .+4
1834  011124  104006                          HLT                             ;SR0 INCORRECT AFTER INIT
1835  011126  005077  167672          DONE2:  CLR     @SR0            ;REINITIALIZE SR0
1836  011132  016777  170030  170024          MOV     KTSTA,@KTVEC
1837  011140  012737  000016  000014          MOV     #16,@#14        ;RESTORE T-BIT TRAP CATCHER
1838
1839                                  ;SHOW THAT INIT CLEARS SR0<0-3,5-6>
```

```
1840                                            ;REFERENCE NR USER PAGE 7 TO SET ALL BITS(0-6)
1841                                            ;THEN ISSUE INIT
1842  011146  104400                           TEST36: SCOPE
1843  011150  012706  001000                           MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1844  011154  004767  005224                           JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1845  011160  000036                                   36                              ;TEST NUMBER
1846  011162  104006                                   HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1847  011164  004767  004224                           JSR     %7,RWALL        ;MAP ALL PAR/PDR PAIR'S INITIALLY RW,4K,
1848                                                                            ;BANK 0
1849  011170  012777  077400  167654                   MOV     #77400,@UPDR7   ;MAKE USER 7 NR
1850  011176  004767  005266                           JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1851  011202  012777  011240  167754                   MOV     #RET3,@KTVEC    ;SETUP ABORT RETURN
1852  011210  005077  167752                           CLR     @KTSTA
1853  011214  012737  140000  177776                   MOV     #140000,@#PS    ;SET MODE TO USER
1854  011222  012706  000400                           MOV     #USTACK,R6      ;SETUP USER STACK IN CASE NEEDED
1855  011226  005277  167572                           INC     @SR0            ;TURN ON MEMORY MANAG.
1856  011232  005737  160000                           TST     @#160000        ;REFERENCE PAGE 7
1857  011236  000777                                   BR      .               ;NO ABORT ON NR REFERENCE
1858  011240  022777  100157  167556  RET3:            CMP     #100157,@SR0    ;CHECK SR0
1859  011246  001401                                   BEQ     .+4
1860  011250  104006                                   HLT                     ;SR0 INCORRECT - SHOULD HAVE TRACKED
1861                                                                            ;NR REFERENCE TO USER 7
1862  011252  000005                                   RESET                   ;ISSUE INIT
1863  011254  005777  167544                           TST     @SR0            ;CHECK SR0
1864  011260  001401                                   BEQ     .+4
1865  011262  104006                                   HLT                     ;SR0 INCORRECT AFTER INIT
1866  011264  005077  167534                           CLR     @SR0
1867  011270  012767  000010  004360                   MOV     #10,ICOUNT      ;DROP ITERATION COUNT
1868  011276  016777  167664  167660                   MOV     KTSTA,@KTVEC
1869
1870                                            ;SHOW THAT BYTE ADDRESSING OF SR0 WORKS
1871  011304  104400                           TEST37: SCOPE
1872  011306  012706  001000                           MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1873  011312  004767  005066                           JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1874  011316  000037                                   37                              ;TEST NUMBER
1875  011320  104006                                   HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1876  011322  004767  004066                           JSR     %7,RWALL        ;MAP ALL PAR/PDR PAIRS RW,4K,BANK 0
1877  011326  004767  005136                           JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1878  011332  012777  160001  167464                   MOV     #160001,@SR0    ;TURN ON MEMORY MANAG. AND SET ERROR FLAGS
1879  011340  105077  167460                           CLRB    @SR0            ;DATOB (LOW) TO SR0
1880  011344  032777  160000  167452                   BIT     #160000,@SR0    ;CHECK SR0
1881  011352  001001                                   BNE     .+4
1882  011354  104006                                   HLT                     ;SR0 INCORRECT AFTER DATOB
1883  011356  012777  160001  167440                   MOV     #160001,@SR0
1884  011364  105077  167436                           CLRB    @SR0H           ;DATOB (HIGH) TO SR0
1885  011370  022777  000017  167426                   CMP     #17,@SR0                ;CHECK SR0
1886  011376  001401                                   BEQ     .+4
1887  011400  104006                                   HLT                     ;SR0 INCORRECT AFTER DATOB
1888  011402  005077  167416                           CLR     @SR0
1889
1890                                            ;SHOW THAT SR0 (1-3) TRACK PAGE REFERENCED IF
1891                                            ;MEMORY MANAG. IS ON AND REFERENCE IS NOT TO A MEMORY MANAG. REGISTER
1892                                            ;SHOW THAT EACH VALUE IS CORRECTLY "LOCKED" IN SR0 AFTER AN ABORT
1893  011406  104400                           TEST40: SCOPE
1894  011410  012706  001000                           MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1895  011414  004767  004764                           JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
```

```
1896  011420  000040                        40                              ;TEST NUMBER
1897  011422  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1898  011424  004767  003764                JSR     %7,RWALL
1899  011430  004767  005034                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1900  011434  012777  011512  167522        MOV     #RET5,@KTVEC
1901  011442  005077  167520                CLR     @KTSTA
1902  011446  016701  167362                MOV     UPDR0,R1
1903  011452  005002                        CLR     R2
1904  011454  012703  100141                MOV     #100141,R3
1905  011460  012704  000010                MOV     #10,R4
1906  011464  012711  077400        LOOP5:  MOV     #77400,@R1                      ;MAKE USER NR
1907  011470  012737  140000  177776        MOV     #140000,@#PS    ;ENTER USER MODE
1908  011476  005277  167322                INC     @SR0
1909  011502  005712                        TST     @R2
1910  011504  000777                        BR      .                       ;REFERENCE TO NR PAGE DIDN'T ABORT
1911  011506  000005                        RESET                           ;AFTER ERROR, TURN OFF MEMORY MANAG.
1912  011510  000423                        BR      DONE5
1913  011512  017705  167306        RET5:   MOV     @SR0,R5                 ;SAVE CONTENTS OF SR0
1914  011516  005077  167302                CLR     @SR0                    ;TURN OFF MEMORY MANAG.
1915  011522  020503                        CMP     R5,R3                   ;CHECK SAVED CONTENTS OF SR0
1916  011524  001401                        BEQ     .+4
1917  011526  104006                        HLT                             ;SR0 INCORRECT
1918  011530  020167  167300                CMP     R1,UPDR0                ;IS USER 0 UNDER TEST
1919  011534  001002                        BNE     LOP5A                   ;NO, CONTINUE
1920  011536  012711  077406                MOV     #77406,@R1              ;MAKE USER 0 RESIDENT
1921  011542  022626                LOP5A:  CMP     (R6)+,(R6)+
1922  011544  005721                        TST     (R1)+
1923  011546  062703  000002                ADD     #2,R3
1924  011552  062702  020000                ADD     #20000,R2
1925  011556  077436                        SOB     R4,LOOP5
1926  011560  016777  167402  167376  DONE5:  MOV     KTSTA,@KTVEC
1927  011566  005077  167374                CLR     @KTSTA
1928
1929                                  ;SHOW THAT SR0 <5-6> TRACK PAGE REFERENCED (MODE) IF
1930                                  ;MEMORY MANAG. IS ON AND THE REFERENCE IS NOT TO A MEMORY MANAG. REGISTER
1931                                  ;SHOW THAT EACH VALUE IS CORRECTLY "LOCKED" IN SR0 AFTER AN ABORT
1932  011572  104400                TEST41: SCOPE
1933  011574  012706  001000                MOV     #KSTACK,SP                      ;INITIALIZE KERNEL STACK POINTER
1934  011600  004767  004600                JSR     PC,ORDER                        ;CHECK TEST SEQUENCE + INIT SR0
1935  011604  000041                        41                                      ;TEST NUMBER
1936  011606  104006                        HLT                                     ;TEST EXECUTED OUT OF SEQUENCE
1937  011610  004767  003600                JSR     %7,RWALL        ;MAP ALL PAGES RW,4K, BANK 0
1938  011614  004767  004650                JSR     PC,KERN7                        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1939  011620  012777  077400  167250        MOV     #77400,@KPDR1   ;SETUP PAGE 1 IN EACH MODE TO BE NR
1940  011626  012777  077400  167202        MOV     #77400,@UPDR1
1941  011634  012777  011662  167322        MOV     #RET7A,@KTVEC   ;SETUP ABORT RETURN
1942  011642  005277  167156                INC     @SR0            ;TURN ON MEMORY MANAG.
1943  011646  005737  020000                TST     @#20000         ;REFERENCE PAGE 1 (NR)
1944  011652  005077  167146                CLR     @SR0            ;TURN OFF MEMORY MANAG.
1945  011656  104006                        HLT                     ;NR REFERENCE DIDN'T ABORT
1946  011660  000436                        BR      DONE7
1947  011662  017701  167136        RET7A:  MOV     @SR0,R1         ;SAVE SR0 CONTENTS IN R1
1948  011666  005077  167132                CLR     @SR0            ;TURN OFF MEMORY MANAG.
1949  011672  022701  100003                CMP     #100003,R1      ;CHECK SAVED CONTENTS OF SR0
1950  011676  001401                        BEQ     .+4
1951  011700  104006                        HLT                     ;SR0 INCORRECT SHOULD SHOW NR ERR, KERNEL PAGE 1
```

```
1952  011702  012777  011736  167254          MOV     #RET7C,@KTVEC   ;SETUP NEXT ABORT RETURN
1953  011710  012737  140000  177776          MOV     #140000,@#PS    ;CHANGE MODE TO USER
1954  011716  005277  167102                  INC     @SR0            ;TURN ON MEMORY MANAG.
1955  011722  005737  020000                  TST     @#20000         ;REFERENCE USER PAGE 1 (NR)
1956  011726  005077  167072                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1957  011732  104006                          HLT                     ;NR REFERENCE DIDN'T ABORT
1958  011734  000410                          BR      DONE7
1959  011736  017701  167062          RET7C:  MOV     @SR0,R1         ;SAVE CONTENTS OF SR0
1960  011742  005077  167056                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
1961  011746  022701  100143                  CMP     #100143,R1      ;CHECK SAVED CONTENTS OF SR0
1962  011752  001401                          BEQ     .+4
1963  011754  104006                          HLT                     ;SR0 INCORRECT - SHOULD SHOW NR
1964                                                                  ;ERROR, USER PAGE 1
1965  011756  016777  167204  167200  DONE7:  MOV     KTSTA,@KTVEC    ;RESTORE TRAP CATCHER
1966
1967                          ;SHOW THAT SR0 (1-3,5-6) DOESN'T TRACK IF MEMORY MANAG. IS OFF BUT DOES IF REFERENCE IS
1968                          ;AN INTERNAL (MEMORY MANAG.) REGISTER
1969  011764  104400          TEST42: SCOPE
1970  011766  012706  001000          MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1971  011772  004767  004406          JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1972  011776  000042                  42                              ;TEST NUMBER
1973  012000  104006                  HLT                             ;TEST EXECUTED OUT OF SEQUENCE
1974  012002  004767  003406          JSR     %7,RWALL        ;SET ALL PAR/PDR PAIRS RW, 4K, BANK 0
1975  012006  012777  007600  167056  MOV     #7600,@UPAR7    ;MAP USER 7 TO THE EXT. BANK
1976  012014  012737  140000  177776  MOV     #140000,@#PS    ;SET MODE TO USER
1977  012022  005277  166776          INC     @SR0            ;TURN ON MEMORY MANAG.
1978  012026  042777  000001  166770  BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
1979  012034  005037  177776          CLR     @#PS            ;CHANGE TO KERNEL MODE
1980  012040  022777  000156  166756  CMP     #156,@SR0       ;CHECK SR0
1981  012046  001401                  BEQ     .+4
1982  012050  104006                  HLT                     ;SR0 INCORRECT - SHOULD SHOW REFERENCE
1983                                                          ;TO USER 7
1984                                                          ;IF IT SHOWS USER 0
1985                                                          ;IT DID NOT TRACK THE INTERNAL REFERENCE
1986                                                          ;IF IT SHOWS KERNEL 0, IT IS
1987                                                          ;TRACKING WITH MEMORY MANAG. OFF
1988  012052  005077  166746          CLR     @SR0
1989
1990                          ;SHOW THAT IF AN INSTRUCTION IS COMPLETED BEFORE A MEMORY MANAGEMENT FAULT
1991                          ;OCCURS, SR2 WILL CONTAIN THE ADDRESS OFLAST FETCH BEFORE ABORT
1992                          ;TO TEST THIS, TRACE TRAP IS USED.  THE VECTOR IS MADE NON-RESIDENT BY MAKING
1993                          ;KERNEL PAGE 0 MAPPED DOWN FROM 17776 TO 100. THUS THE MEMORY MANAGEMENT
1994                          ;VECTOR IS RESIDENT WHILE THE TRACE TRAP VECTOR IS OUTSIDE THE ALLOWED
1995                          ;PAGE LENGTH.
1996  012056  104400          TEST43: SCOPE
1997  012060  012706  001000          MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
1998  012064  004767  004314          JSR     %C,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
1999  012070  000043                  43                              ;TEST NUMBER
2000  012072  104006                  HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2001  012074  004767  003314          JSR     %7,RWALL        ;INITIALIZE ALL PAGES RW,4K,BANK 0
2002  012100  012777  000416  166766  MOV     #416,@KPDR0     ;MAP KERNEL TO EXCLUDE
2003                                                          ;LOCATIONS 0 TO 77
2004  012106  004767  004356          JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2005  012112  012777  012150  167044  MOV     #RET11,@KTVEC   ;SETUP MEMORY MANAGEMENT ABORT RETURN
2006  012120  005077  167042          CLR     @KTSTA
2007  012124  012746  000020          MOV     #20,-(SP)       ;PREPARE STACK TO TURN ON T-BIT
```

```
2008  012130  012746  012136                    MOV     #.+6,-(SP)
2009  012134  000006                            RTT
2010  012136  012777  000001  166660  ADR11:    MOV     #1,@SR0         ;SET T-BIT VIA RTT
2011                                                                    ;TURN ON MEMORY MANAG. - SHOULD
2012                                                                    ;ATTEMPT TO TRACE TRAP AT END OF
2013                                                                    ;INSTRUCTION - SHOULD GET A PAGE
2014  012144  000000                            HALT                    ;LENGTH ERROR ON THAT ATTEMPT
2015  012146  000415                            BR      CONT11          ;NO PAGE LENGTH ERROR ON TRACE TRAP
2016  012150  042777  000001  166646  RET11:    BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
2017  012156  022777  040000  166640            CMP     #40000,@SR0     ;CK SR0
2018  012164  001401                            BEQ     .+4
2019  012166  104006                            HLT                     ;SR0 INCORRECT - PL FAULT,KERNEL 0 REFERENCE COMPLETED
2020  012170  022777  012136  166634            CMP     #ADR11,@SR2     ;CK SR2
2021  012176  001401                            BEQ     .+4
2022  012200  104006                            HLT                     ;SR2 INCORRECT - SHOULD CONTAIN
2023                                                                    ;ADDRESS OF  LAST FETCH BEFORE ABORT
2024  012202  005077  166616          CONT11:   CLR     @SR0            ;REINITIALIZE SR0
2025  012206  016777  166754  166750            MOV     KTSTA,@KTVEC    ;RESTORE TRAP CATCHER
2026
2027                            ;SHOW THAT HAVING THE ABORT ERROR
2028                            ;BITS SET WILL NOT PREVENT A MEMORY MANAGEMENT TRAP
2029  012214  104400            TEST44: SCOPE
2030  012216  012706  001000                    MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2031  012222  004767  004156                    JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2032  012226  000044                            44                              ;TEST NUMBER
2033  012230  104006                            HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2034  012232  004767  003156                    JSR     X7,RWALL        ;INITIALIZE ALL PAR/PDR PAIRS TO RW,4K, BANK 0
2035  012236  012777  077402  166634            MOV     #77402,@KPDR2   ;SET KERNEL PAR/PDR PAIR 2 RR0,4K
2036  012244  004767  004220                    JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2037  012250  012777  012312  166706            MOV     #RET13A,@KTVEC  ;SETUP MEMORY MANAGEMENT ABORT RETURN
2038  012256  005077  166704                    CLR     @KTSTA
2039  012262  005277  166536                    INC     @SR0            ;TURN ON MEMORY MANAG.
2040  012266  012777  160001  166530            MOV     #160001,@SR0    ;SET ABORT ERROR BITS
2041  012274  013737  007000  047000            MOV     @#7000,@#47000  ;WRITE KERNEL PAR/PDR PAIR 2 (RR0)-SHOULD TRAP
2042  012302  005077  166516                    CLR     @SR0
2043  012306  104006                            HLT                     ;NO TRAP OCCURRED
2044  012310  000416                            BR      DONE13
2045  012312  022626          RET13A:   CMP     (SP)+,(SP)+     ;RESTORE THE STACK POINTER
2046  012314  017701  166504                    MOV     @SR0,R1         ;SAVE CONTENTS OF SR0
2047  012320  005077  166500                    CLR     @SR0            ;TURN OFF MEMORY MANAG.
2048  012324  022701  160017                    CMP     #160017,R1
2049  012330  001401                            BEQ     .+4
2050  012332  104006                            HLT                     ;SAVED CONTENTS OF SR0 INCORRECT
2051  012334  022777  077402  166536            CMP     #77402,@KPDR2   ;CHECK THE PDR CORRESPONDING TO THE TRAP REFERENCE
2052  012342  001401                            BEQ     .+4
2053  012344  104006                            HLT                     ;THE PDR CORRESPONDING TO THE TRAP REFERENCE IS INCORREC
2054  012346  016777  166614  166610  DONE13:   MOV     KTSTA,@KTVEC    ;RESTORE MEMORY MANAGEMENT TRAP RETURN
2055                                                                    ;TO CAUSE A HALT ON A FALSE TRAP OR ABORT
2056  012354  005077  166444                    CLR     @SR0            ;REINITIALIZE SR0
2057
2058                            ;SHOW THAT MEMORY MANAGEMENT WILL NOT TRAP ON AN INTERNAL REFERENCE
2059  012360  104400            TEST45: SCOPE
2060  012362  012706  001000                    MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2061  012366  004767  004012                    JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2062  012372  000045                            45                              ;TEST NUMBER
2063  012374  104006                            HLT                             ;TEST EXECUTED OUT OF SEQUENCE
```

```
2064  012376  004767  003012                    JSR     X7,RWALL        ;MAP ALL PAR/PDR PAIRS 4K, RW, BANK 0
2065  012402  004767  004062                    JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2066  012406  012777  012450  166550            MOV     #RET16,@KTVEC   ;SETUP TRAP RETURN IN CASE
2067  012414  005077  166546                    CLR     @KISTA
2068  012420  005277  166400                    INC     @SR0            ;TURN ON MEMORY MANAG.
2069  012424  005777  166374                    TST     @SR0            ;TRAP REFERENCE TO A MEMORY MANAG. REGISTER
2070  012430  005077  166370                    CLR     @SR0
2071  012434  022777  077406  166450            CMP     #77406,@KPDR7
2072  012442  001401                            BEQ     .+4
2073  012444  104006                            HLT
2074  012446  000404                            BR      DONE16
2075  012450  042777  000001  166346  RET16:    BIC     #1,@SR0
2076  012456  104006                            HLT
2077  012460  005077  166340          DONE16:   CLR     @SR0
2078  012464  016777  166476  166472            MOV     KTSTA,@KTVEC
2079
2080                                  ;TEST PAGE LENGTH ERROR CHECKING (EXPAND DOWN NOT SET)
2081                                  ;KERNEL PAR/PDR PAIR1 IS USED WITH ALL PAGE LENGTH VALUES
2082                                  ;SHOW THAT REFERENCES TO BOTH BOUNDARIES OF THE ALLOWED AREA DON'T TRAP OR ABORT
2083                                  ;SHOW THAT A REFERENCE TO THE FIRST WORD BEYOND THE ALLOWABLE AREA DOES TRAP
2084  012472  104400                  TEST46: SCOPE
2085  012474  012706  001000                    MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2086  012500  004767  003700                    JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2087  012504  000046                            46                              ;TEST NUMBER
2088  012506  104006                            HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2089  012510  004767  002700                    JSR     X7,RWALL        ;INITIALIZE ALL PAR/PDR PAIRS TO RW,4K, BANK  0
2090  012514  004767  003750                    JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2091  012520  012702  000006                    MOV     #6,R2           ;R2 CONTAINS VALUE TO BE LOADED IN THE
2092                                                                    ;PDR BEING CHECKED (INCLUDING PLF)
2093  012524  012701  020076                    MOV     #20076,R1       ;R1 IS USED TO REFERENCE THE TOP ADDRESS
2094                                                                    ;WITHIN THE ALLOWED AREA
2095  012530  012777  012610  166426            MOV     #RET23A,@KTVEC  ;SETUP ABORT RETURN IN CASE REFERENCE
2096  012536  005077  166424                    CLR     @KTSTA          ;WITHIN ALLOWED AREA ABORTS
2097  012542  005277  166256          LOOP23:   INC     @SR0            ;TURN ON MEMORY MANAG.
2098  012546  010277  166324                    MOV     R2,@KPDR1       ;SET KERNEL PAR/PDR PAIR 1 TO NEW PAGE LENGTH
2099  012552  005727  020000                    TST     #20000          ;READ LOWER BOUNDARY-SHOULDN'T ABORT
2100  012556  005711                            TST     @R1             ;READ UPPER ALLOWED BOUNDARY-SHOULDN'T
2101                                                                    ;ABORT
2102  012560  012777  012630  166376            MOV     #RET23B,@KTVEC  ;SETUP ABORT RETURN
2103  012566  020127  037776                    CMP     R1,#37776       ;CHECK FOR DONE (TO AVOID REFERENCING
2104                                                                    ;NEXT PAR/PDR PAIR)
2105  012572  103041                            BHIS    DONE23          ;EXIT LOOP IF DONE
2106  012574  005761  000002                    TST     2(R1)           ;REFERENCE OUTSIDE ALLOWED AREA -
2107                                                                    ;SHOULD ABORT
2108  012600  005077  166220                    CLR     @SR0            ;TURN MEMORY MANAG. OFF
2109  012604  104006                            HLT                     ;NO ABORT OCCURRED ON A REFERENCE
2110  012606  000426                            BR      CONT23          ;OUTSIDE THE ALLOWED PAGE LENGTH
2111  012610  042777  000001  166206  RET23A:   BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
2112  012616  022626                            CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
2113  012620  104006                            HLT                     ;REFERENCE WITHIN ALLOWED AREA
2114  012622  005077  166176                    CLR     @SR0            ;CLEAR ERROR BITS
2115  012626  000416                            BR      CONT23          ;CAUSED A TRAP OR ABORT
2116  012630  022626                  RET23B:   CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
2117  012632  017703  166166                    MOV     @SR0,R3         ;SAVE CURRENT SR0
2118  012636  005077  166162                    CLR     @SR0            ;TURN OFF MEMORY MANAG.
2119  012642  022703  040003                    CMP     #40003,R3       ;CK SAVED SR0
```

```
2120  012646  001401                          BEQ     .+4
2121  012650  104006                          HLT                     ;CONTENTS OF SR0 INCORRECT AFTER
2122                                                                  ;PAGE LENGTH ERROR ABORT
2123  012652  022777  000002  166144          CMP     #2,@SR0         ;CHECK SR0 TO BE SURE PL BIT CLEARED
2124  012660  001401                          BEQ     .+4
2125  012662  104006                          HLT                     ;SR0 INCORRECT AFTER CLEARING IT
2126                                                                  ;ONLY KERNEL PAGE 1 SHOULD STILL BE SET
2127  012664  062701  000100        CONT23: ADD     #100,R1         ;SETUP R1 TO REFERENCE BOUNDARY OF
2128                                                                  ;NEXT PAGE
2129  012670  062702  000400                ADD     #400,R2         ;ADD 1 TO VALUE TO BE LOADED IN
2130                                                                  ;PAGE LENGTH FIELD
2131  012674  000722                  BR      LOOP23          ;CHECK NEXT PAGE LENGTH VALUE
2132  012676  005077  166122        DONE23: CLR     @SR0            ;TURN OFF MEMORY MANAG.
2133  012702  016777  166260  166254        MOV     KTSTA,@KTVEC    ;RESTORE MEMORY MANAGEMENT ABORT RETURN
2134  012710  005077  166252                CLR     @KTSTA          ;TO CAUSE HALT ON A FALSE TRAP
2135                                                                  ;OR ABORT
2136
2137                          ;TEST PAGE LENGTH ERROR CHECKING (EXPAND DOWN SET)
2138                          ;KERNEL PAR/PDR PAIR1 IS TESTED WITH ALL VALUES OF PAGE LENGTH FIELD
2139                          ;SHOW THAT REFERENCES TO BOTH BOUNDARIES OF THE ALLOWED AREA DON'T TRAP OR ABORT
2140                          ;SHOW THAT A REFERENCE TO THE WORD IMMEDIATELY BELOW THE ALLOWED AREA DOES TRAP
2141  012714  104400        TEST47: SCOPE
2142  012716  012706  001000        MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2143  012722  004767  003456        JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2144  012726  000047                47                              ;TEST NUMBER
2145  012730  104006                HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2146  012732  004767  002456        JSR     %7,RWALL        ;INITIALIZE ALL PAR/PDR PAIRS TO RW,4K, BANK 0
2147  012736  004767  003526        JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2148  012742  012702  077416        MOV     #77416,R2       ;R2 CONTAINS VALUE TO BE LOADED IN THE
2149                                                                  ;PDR BEING CHECKING (INCLUDING PLF)
2150  012746  012701  037700        MOV     #37700,R1       ;R1 IS USED TO REFERENCE THE LOWEST
2151                                                                  ;ALLOWED ADDRESS IN THE PAGE
2152  012752  012777  013032  166204        MOV     #RET24A,@KTVEC  ;SETUP ABORT RETURN IN CASE REFERENCE
2153  012760  005077  166202                CLR     @KTSTA          ;WITHIN ALLOWED AREA ABORTS
2154  012764  005277  166034        LOOP24: INC     @SR0            ;TURN ON MEMORY MANAG.
2155  012770  010277  166102                MOV     R2,@KPDR1       ;SET KERNEL PAR/PDR PAIR 1 TO NEW PAGE LENGTH
2156  012774  005727  037776                TST     #37776          ;REFERENCE UPPER ALLOWED BOUNDARY
2157  013000  005711                        TST     @R1             ;REFERENCE LOWER ALLOWED BOUNDARY
2158                                                                  ;- NEITHER REFERENCE SHOULD ABORT
2159  013002  012777  013044  166154        MOV     #RET24B,@KTVEC  ;SETUP ABORT RETURN
2160  013010  020127  020000                CMP     R1,#20000       ;CHECK FOR DONE
2161  013014  001436                        BEQ     DONE24          ;EXIT LOOP IF DONE
2162  013016  005761  177776                TST     -2(R1)          ;REFERENCE BELOW ALLOWED AREA -
2163                                                                  ;SHOULD ABORT
2164  013022  005077  165776                CLR     @SR0            ;TURN MEMORY MANAG. OFF
2165  013026  104006                        HLT                     ;NO ABORT OCCURRED ON A REFERENCE
2166  013030  000423                        BR      CONT24          ;OUTSIDE THE ALLOWED PAGE LENGTH
2167  013032  005077  165766        RET24A: CLR     @SR0            ;TURN OFF MEMORY MANAG. AND CLEAR
2168                                                                  ;ERROR BITS
2169  013036  022626                        CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
2170  013040  104006                        HLT                     ;REFERENCE WITHIN ALLOWED AREA CAUSED
2171  013042  000416                        BR      CONT24          ;A TRAP OR ABORT
2172  013044  022626        RET24B: CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
2173  013046  017703  165752                MOV     @SR0,R3         ;SAVE CURRENT SR0
2174  013052  005077  165746                CLR     @SR0            ;TURN OFF MEMORY MANAG.
2175  013056  022703  040003                CMP     #40003,R3       ;CK SAVED SR0
```

```
2176  013062  001401                        BEQ     .+4
2177  013064  104006                        HLT                     ;CONTENTS OF SR0 INCORRECT AFTER
2178                                                                  ;PAGE LENGTH ERROR ABORT
2179  013066  022777  000002  165730        CMP     #2,@SR0         ;CHECK SR0 TO BE SURE PL BIT CLEARED
2180  013074  001401                        BEQ     .+4
2181  013076  104006                        HLT                     ;SR0 INCORRECT AFTER CLEARING IT
2182  013100  162701  000100  CONT24:      SUB     #100,R1         ;SETUP R1 TO REFERENCE BOUNDARY
2183                                                                  ;OF NEXT PAGE DOWN
2184  013104  162702  000400                SUB     #400,R2         ;INCREASE ALLOWED PAGE LENGTH
2185                                                                  ;(DOWN) BY 1 PAGE
2186  013110  000725                        BR      LOOP24          ;CHECK NEXT PAGE LENGTH VALUE
2187  013112  005077  165706  DONE24:      CLR     @SR0            ;TURN OFF MEMORY MANAG.
2188  013116  016777  166044  166040        MOV     KTSTA,@KTVEC    ;RESTORE MEMORY MANAGEMENT ABORT RETURN
2189  013124  005077  166036                CLR     @KTSTA          ;TO CAUSE A HALT ON A FALSE TRAP
2190                                                                  ;OR ABORT
2191
2192                                ;TEST ALL COMBINATIONS OF VALUES FOR THE PAGE LENGTH COMPARATORS-
2193                                ;USE KERNEL PAGE PAGE 1
2194  013130  104400                TEST50: SCOPE
2195  013132  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2196  013136  004767  003242                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2197  013142  000050                        50                              ;TEST NUMBER
2198  013144  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2199  013146  012767  000020  002502        MOV     #20,ICOUNT      ;DROP ITERATION COUNT
2200  013154  004767  002234                JSR     X7,RWALL        ;INITIALIZE ALL PAGES RW, BANK 0
2201  013160  004767  003304                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2202  013164  012777  013276  165772        MOV     #RET25,@KTVEC   ;SETUP ABORT RETURN
2203  013172  005077  165770                CLR     @KTSTA
2204  013176  012701  000006                MOV     #6,R1           ;R1 CONTAINS THE VALUE TO BE
2205                                                                  ;LOADED INTO THE PDR
2206  013202  012777  000001  165614        MOV     #1,@SR0         ;TURN ON MEMORY MANAG.
2207  013210  012703  020000  L25A:         MOV     #20000,R3       ;R3 CONTAINS VA USED
2208  013214  010177  165656                MOV     R1,@KPDR1       ;LOAD NEW PAGE LENGTH FIELD
2209  013220  010102  L25B:                 MOV     R1,R2           ;R2 IS A COPY OF R1
2210  013222  010304                        MOV     R3,R4           ;R4 IS A COPY OF R3
2211  013224  042704  160000                BIC     #160000,R4
2212  013230  005713                        TST     (R3)            ;USE VA IN R3 TO REFERENCE PAGE 1
2213  013232  000302                        SWAB    R2              ;NO TRAP-CHECK TO MAKE SURE
2214  013234  042702  177400                BIC     #177400,R2
2215  013240  006304                        ASL     R4              ;VIRTUAL ADDRESS WAS WITHIN
2216  013242  006304                        ASL     R4
2217  013244  000304                        SWAB    R4
2218  013246  020402                        CMP     R4,R2           ;ALLOWED PAGE LENGTH
2219  013250  003401                        BLE     .+4
2220  013252  104006                        HLT                     ;REFERENCE OUTSIDE ALLOWED PAGE LENGTH
2221                                                                  ;DIDN'T ABORT
2222  013254  062703  000100  C25:          ADD     #100,R3
2223  013260  020327  037776                CMP     R3,#37776
2224  013264  003755                        BLE     L25B
2225  013266  062701  000400                ADD     #400,R1
2226  013272  100346                        BPL     L25A
2227  013274  000413                        BR      DONE25
2228  013276  022626  RET25:               CMP     (SP)+,(SP)+     ;RESTORE STACK POINTER
2229  013300  000302                        SWAB    R2              ;CHECK TO MAKE SURE VIRTUAL
2230  013302  042702  177400                BIC     #177400,R2
2231  013306  006304                        ASL     R4              ;ADDRESS WAS OUTSIDE ALLOWED
```

```
2232  013310  006304                          ASL     R4
2233  013312  000304                          SWAB    R4
2234  013314  020402                          CMP     R4,R2           ;PAGE LENGTH
2235  013316  003001                          BGT     .+4
2236  013320  104006                          HLT                     ;REFERENCE WITHIN ALLOWED
2237  013322  000754                          BR      C25             ;PAGE LENGTH ABORTED-R3 CONTAINS
2238                                                                  ;VA USED, R1 CONTAINS VALUE
2239                                                                  ;LOADED INTO THE PDR
2240  013324  016777  165636  165632  DONE25: MOV     KTSTA,@KTVEC
2241  013332  005077  165466                  CLR     @SR0
2242
2243                          ;SHOW THAT THE W BIT DOESN'T SET IF THE MEMORY MANAG. IS OFF
2244  013336  104400          TEST51: SCOPE
2245  013340  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2246  013344  004767  003034                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2247  013350  000051                          51                              ;TEST NUMBER
2248  013352  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2249  013354  012767  002000  002274          MOV     #2000,ICOUNT    ;RESTORE ITERATION COUNT
2250  013362  004767  002002                  JSR     %7,CLRALL       ;CLEAR ALL MEMORY MANAG. REGISTERS
2251  013366  013737  010000  010000          MOV     @#10000,@#10000 ;WRITE BANK 0
2252  013374  005777  165474                  TST     @KPDR0
2253  013400  001401                          BEQ     .+4
2254  013402  104006                          HLT                     ;W BIT SET OR ANOTHER BIT INCORRECT
2255                                                                  ;IN KERNEL 0 PDR
2256
2257                          ;SHOW THAT THE W BIT IS CLEARED BY WRITING (VIA DATO) THE CORRESPONDING PAR
2258                          ;CHECK EACH PDR
2259  013404  104400          TEST52: SCOPE
2260  013406  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2261  013412  004767  002766                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2262  013416  000052                          52                              ;TEST NUMBER
2263  013420  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2264  013422  004767  001766                  JSR     %7,RWALL
2265  013426  004767  003036                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2266  013432  012777  007600  165432          MOV     #7600,@UPAR7    ;MAP USER 7 TO EXTERNAL BANK
2267  013440  012737  140000  177776          MOV     #140000,@#PS    ;SET MODE TO USER
2268  013446  012706  000400                  MOV     #USTACK,R6      ;SETUP USER STACK
2269  013452  012700  001144                  MOV     #STATAB,R0      ;SET UP KT REG TABLE POINTER
2270  013456  012001          LOP27:  MOV     (R0)+,R1        ;R1 CONTAINS ADDRESS OF
2271                                                                  ;ADDRESS OF CURRENT PDR
2272  013460  012702  017776                  MOV     #17776,R2       ;R2 CONTAINS VIRTUAL ADDRESS TO
2273                                                                  ;REFERENCE DESIRED PAGE
2274  013464  012037  177776                  MOV     (R0)+,@#PS      ;SETUP STATUS FOR CURRENT MODE
2275  013470  005277  165330          LOP27A: INC     @SR0            ;TURN ON MEMORY MANAG.
2276  013474  011212                          MOV     (R2),(R2)       ;WRITE
2277  013476  005077  165322                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
2278  013502  004767  000016                  JSR     %7,CKWBIT       ;TEST W BIT
2279  013506  062702  020000                  ADD     #20000,R2       ;CHANGE VA TO REFERENCE NEXT PAGE
2280  013512  103366                          BCC     LOP27A          ;LOOP UNTIL ALL PDR'S HAVE BEEN
2281                                                                  ;CHECKED IN THE CURRENT MODE
2282  013514  020027  001152                  CMP     R0,#STAEND
2283  013520  002756                          BLT     LOP27
2284  013522  000416                          BR      EXT27
2285  013524  032771  000100  000000  CKWBIT: BIT     #100,@(R1)      ;CHECK W BIT
2286  013532  001001                          BNE     .+4
2287  013534  104006                          HLT                     ;W BIT DIDN'T SET IN PDR WHOSE
```

```
2288
2289  013536  005071  000020                CLR     @20(R1)         ;ADDRESS IS POINTED TO BY R1
2290                                                                  ;CLEAR W BIT BY WRITING CORRESPONDING
2291  013542  032771  000100  000000        BIT     #100,@(R1)      ;PAR VIA DATO
2292  013550  001401                        BEQ     .+4             ;CHECK W BIT
2293  013552  104006                        HLT                     ;W BIT DIDN'T CLEAR IN PDR WHOSE
2294  013554  005721                        TST     (R1)+           ;ADDRESS IS POINTED TO BY R1
2295  013556  000207                        RTS     X7
2296  013560                        EXT27:
2297
2298                                ;SHOW THAT THE W BIT IS CLEARED BY A DATOB TO THE PDR
2299                                ;CHECK BOTH HIGH AND LOW DATOB'S, ON KERNEL 0
2300  013560  104400                TEST53: SCOPE
2301  013562  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2302  013566  004767  002612                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2303  013572  000053                        53                              ;TEST NUMBER
2304  013574  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2305  013576  004767  001612                JSR     X7,RWALL        ;MAP ALL PAR/PDR PAIRS 4K, RW, BANK 0
2306  013602  004767  002662                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2307  013606  005277  165212                INC     @SR0            ;TURN ON MEMORY MANAG.
2308  013612  013737  000000  000000        MOV     @#0,@#0         ;WRITE INTO PAGE 0
2309  013620  005077  165200                CLR     @SR0            ;TURN OFF MEMORY MANAG.
2310  013624  032777  000100  165242        BIT     #100,@KPDR0     ;CHECK W BIT
2311  013632  001001                        BNE     .+4
2312  013634  104006                        HLT                     ;W BIT NOT SET AFTER WRITING PAGE
2313  013636  112777  000106  165230        MOVB    #106,@KPDR0     ;DATOB SHOULD CLEAR W BIT
2314  013644  032777  000100  165222        BIT     #100,@KPDR0
2315  013652  001401                        BEQ     .+4
2316  013654  104006                        HLT                     ;W BIT DIDN'T CLEAR VIA DATOB (LOW)
2317                                                                  ;TO THE PDR
2318  013656  005277  165142                INC     @SR0            ;TURN ON MEMORY MANAG.
2319  013662  013737  017776  017776        MOV     @#17776,@#17776 ;WRITE INTO PAGE 0 AGAIN
2320  013670  005077  165130                CLR     @SR0            ;TURN OFF MEMORY MANAG.
2321  013674  032777  000100  165172        BIT     #100,@KPDR0     ;CHECK W BIT
2322  013702  001001                        BNE     .+4
2323  013704  104006                        HLT                     ;W BIT NOT SET AFTER WRITING PAGE
2324  013706  016701  165162                MOV     KPDR0,R1        ;SETUP R1 TO REFERENCE HIGH BYTE
2325  013712  005201                        INC     R1              ;OF KPDR0
2326  013714  112711  000177                MOVB    #177,@R1        ;DATOB TO HIGH BYTE OF KPDR0
2327  013720  032777  000100  165146        BIT     #100,@KPDR0     ;CHECK W BIT
2328  013726  001401                        BEQ     .+4
2329  013730  104006                        HLT                     ;W BIT DIDN'T CLEAR VIA DATOB
2330                                                                  ;TO HIGH BYTE OF PDR
2331                                ;SHOW THAT THE W BIT IS CLEARED BY A DATOB TO THE PAR
2332                                ;CHECK BOTH HIGH AND LOW DATOB'S, ON KERNEL 0
2333  013732  104400                TEST54: SCOPE
2334  013734  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2335  013740  004767  002440                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2336  013744  000054                        54                              ;TEST NUMBER
2337  013746  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2338  013750  004767  001440                JSR     X7,RWALL        ;MAP ALL PAR/PDR PAIRS 4K, RW, BANK 0
2339  013754  004767  002510                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2340                                                                  ;EXTERNAL BANK
2341  013760  005277  165040                INC     @SR0            ;TURN ON MEMORY MANAG.
2342  013764  013737  000000  000000        MOV     @#0,@#0         ;WRITE INTO PAGE 0
2343  013772  005077  165026                CLR     @SR0            ;TURN OFF MEMORY MANAG.
```

```
2344  013776  032777  000100  165070          BIT     #100,@KPDR0     ;CHECK W BIT
2345  014004  001001                          BNE     .+4
2346  014006  104006                          HLT                     ;W BIT NOT SET AFTER WRITING PAGE
2347  014010  112777  000000  165076          MOVB    #0,@KPAR0       ;DATOB TO THE PAR
2348  014016  032777  000100  165050          BIT     #100,@KPDR0     ;CHECK W BIT
2349  014024  001401                          BEQ     .+4
2350  014026  104006                          HLT                     ;W BIT DIDN'T CLEAR VIA DATOB
2351                                                                  ;(LOW) TO THE PAR
2352  014030  005277  164770                  INC     @SR0            ;TURN ON MEMORY MANAG.
2353  014034  013737  017776  017776          MOV     @#17776,@#17776 ;WRITE INTO PAGE 0 AGAIN
2354  014042  005077  164756                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
2355  014046  032777  000100  165020          BIT     #100,@KPDR0     ;CHECK W BIT
2356  014054  001001                          BNE     .+4
2357  014056  104006                          HLT                     ;W BIT NOT SET AFTER WRITING PAGE
2358  014060  016701  165030                  MOV     KPAR0,R1        ;SETUP R1 TO REFERENCE HIGH BYTE
2359  014064  005201                          INC     R1              ;OF KPAR0
2360  014066  112711  000000                  MOVB    #0,@R1          ;DATOB TO HIGH BYTE OF KPAR0
2361  014072  032777  000100  164774          BIT     #100,@KPDR0     ;CHECK W BIT
2362  014100  001401                          BEQ     .+4
2363  014102  104006                          HLT                     ;W BIT DIDN'T CLEAR VIA DATOB
2364                                                                  ;TO HIGH BYTE OF PAR
2365
2366                                  ;SHOW THAT THE W BIT IS NOT CLEARED BY INIT
2367                                  ;INITIALLY SET ALL THE W BITS, THEN DO A RESET AND CHECK THE W BITS
2368  014104  104400                  TEST55: SCOPE
2369  014106  012706  001000                  MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2370  014112  004767  002266                  JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2371  014116  000055                          55                              ;TEST NUMBER
2372  014120  104006                          HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2373  014122  012767  000020  001526          MOV     #20,ICOUNT
2374  014130  004767  001260                  JSR     %7,RWALL        ;INITIALIZE ALL PAGES RW,4K,BANK 0
2375  014134  004767  002330                  JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2376  014140  012777  007600  164724          MOV     #7600,@UPAR7    ;MAP USER 7 TO THE EXTERNAL BANK
2377  014146  012737  140000  177776          MOV     #140000,@#PS    ;SET MODE TO USER
2378  014154  012706  000400                  MOV     #USTACK,R6      ;SETUP USER STACK
2379  014160  012700  001144                  MOV     #STATAB,R0      ;R0 POINTS TO INFORMATION FOR
2380                                                                  ;CURRENT MODE
2381  014164  005720                  LOOP32: TST     (R0)+           ;MOVE POINTER
2382  014166  012037  177776                  MOV     (R0)+,@#PS      ;SETUP MODE TO REFERENCE NEXT SET OF REGS
2383  014172  012702  017776                  MOV     #17776,R2       ;SETUP R2 TO REFERENCE DESIRED PAGE
2384  014176  005277  164622                  INC     @SR0
2385  014202  011212                  LOP32C: MOV     (R2),(R2)       ;WRITE IN
2386  014204  062702  020000                  ADD     #20000,R2       ;CHANGE VA TO REFERENCE NEXT PAGE
2387  014210  103374                          BCC     LOP32C          ;SET ALL W-BITS IN CURRENT MODE
2388  014212  005077  164606                  CLR     @SR0            ;TURN OFF MEMORY MANAG.
2389  014216  020027  001152                  CMP     R0,#STAEND      ;CHECK FOR DONE SETTING THE W BITS
2390  014222  002760                          BLT     LOOP32          ;IF NOT, LOOP TO DO NEXT MODE
2391  014224  012701  001034                  MOV     #ADRTAB,R1      ;SETUP R1 TO REFERENCE ADDRESSES OF PDR'S OF PDR'S
2392  014230  012702  000010          LOP32D: MOV     #10,R2          ;USE R2 AS COUNTER TO CHANGE ADDRESS
2393                                                                  ;AT END OF EACH SET OF REGISTERS
2394  014234  032771  000100  000000  LOP32E: BIT     #100,@(R1)      ;CHECK W BIT
2395  014242  001001                          BNE     .+4
2396  014244  104006                          HLT                     ;W BIT NOT SET IN PDR WHOSE
2397                                                                  ;ADDRESS IS POINTED TO BY R1-
2398                                                                  ;SHOULD HAVE BEEN SET WHEN
2399                                                                  ;PAGE WAS WRITTEN INTO
```

```
2400  014246  005721                        TST     (R1)+           ;MOVE POINTER
2401  014250  077207                        SOB     R2,LOP32E       ;CHECK ALL PDR'S IN THIS SET
2402  014252  062701  000020                ADD     #20,R1          ;CHANGE R1 TO REFERENCE NEXT
2403                                                                  ;SET OF PDR ADDRESSES
2404  014256  020127  001132                CMP     R1,#ADREND      ;CHECK FOR DONE
2405  014262  002762                        BLT     LOP32D          ;IF NOT, CHECK NEXT SET OF PDR'S
2406  014264  005037  177776                CLR     @#PS            ;SET MODE TO KERNEL
2407  014270  005277  164530                INC     @SR0            ;TURN MEMORY MANAG. ON
2408  014274  000005                        RESET                   ;INIT WITH MEMORY MANAG. ON
2409  014276  000005                        RESET                   ;INIT WITH MEMORY MANAG. OFF
2410  014300  012701  001034                MOV     #ADRTAB,R1      ;R1 REFERENCES ADDRESS OF PDR
2411  014304  012702  000010        LOP32F: MOV     #10,R2          ;R2 KEEPS TRACK OF WHEN TO CHANGE
2412                                                                  ;REGISTER SETS
2413  014310  032771  000100  000000 LOP32G: BIT     #100,@(R1)      ;CHECK W BIT
2414  014316  001001                        BNE     .+4
2415  014320  104006                        HLT                     ;INIT CLEARED W BIT IN PDR WHOSE
2416                                                                  ;ADDRESS IS POINTED TO BY R1
2417  014322  005721                        TST     (R1)+           ;MOVE POINTER
2418  014324  077207                        SOB     R2,LOP32G       ;CHECK ALL PDR'S IN THIS SET
2419  014326  062701  000020                ADD     #20,R1          ;CHANGE R1 TO REFERENCE NEXT SET
2420                                                                  ;OF PDR ADDRESSES
2421  014332  020127  001132                CMP     R1,#ADREND      ;CHECK FOR DONE
2422  014336  002762                        BLT     LOP32F          ;IF NOT, CHECK NEXT SET OF PDR'S
2423  014340  005077  164460                CLR     @SR0            ;REINITIALIZE SR0
2424
2425                                ;SHOW THAT A DATO TO A PDR WILL CLEAR THE W BIT
2426                                ;EVEN WHEN THE INSTRUCTION ALSO CAUSES A TRAP REFERENCE TO
2427                                ;THE CORRESPONDING PAGE
2428                                ;MAP KERNEL PAGE 1 RRW AND MAKE A WRITE ACCESS TO PAGE 1
2429                                ;TO SET THE W BIT
2430                                ;THEN LOAD THE PDR, MAKING A TRAP REFERENCE TO PAGE 1 IN THE SOURCE
2431                                ;FETCH OF THE SAME INSTRUCTION-THE W BIT SHOULD BE CLEARED DUE
2432                                ;TO THE DATO TO THE PDR
2433  014344  104400                TEST56: SCOPE
2434  014346  012706  001000                MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2435  014352  004767  002026                JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2436  014356  000056                        56                              ;TEST NUMBER
2437  014360  104006                        HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2438  014362  012767  002000  001266        MOV     #2000,ICOUNT
2439  014370  004767  001020                JSR     %7,RWALL        ;INITIALIZE ALL PAGES RW, BANK 0
2440  014374  004767  002070                JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2441  014400  012777  077406  164470        MOV     #77406,@KPDR1   ;MAKE KERNEL PAGE 1 RRW
2442  014406  012777  000001  164410        MOV     #1,@SR0         ;TURN ON MEMORY MANAG.
2443  014414  013737  020000  020000        MOV     @#20000,@#20000 ;READ AND WRITE PAGE 1
2444  014422  022777  077506  164446        CMP     #77506,@KPDR1   ;CHECK PDR OF PAGE 1
2445  014430  001401                        BEQ     .+4
2446  014432  104006                        HLT                     ;KERNEL PAGE 1 PDR
2447                                                                  ;INCORRECT - W BIT SHOULD
2448                                                                  ;BE SET DUE TO PREVIOUS MOVE INSTRUCTION
2449  014434  012767  077506  164360        MOV     #77506,TEMP     ;LOAD TEMP WITH VALUE TO BE MOVED TO KPDR1
2450  014442  016777  004354  164426        MOV     TEMP+20000,@KPDR1       ;PAGE 1 REFERENCE SHOULD SET
2451                                                                  ;BUT DATO TO THE PDR CLEARS W BIT
2452  014450  022777  077406  164420        CMP     #77406,@KPDR1   ;CHECK PAGE 1 PDR
2453  014456  001401                        BEQ     .+4
2454  014460  104006                        HLT                     ;PDR INCORRECT - W BIT
2455  014462  005077  164336                CLR     @SR0            ;SHOULD HAVE BEEN CLEARED
```

```
2456
2457                                    ;CHECK TO SEE THAT MULTIPLE ACCESSES TO A PAGE AFTER SETTING THE
2458                                    ;W BIT DON'T CLEAR THE W BIT
2459  014466  104400                    TEST57: SCOPE
2460  014470  012706  001000                    MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2461  014474  004767  001704                    JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2462  014500  000057                            57                              ;TEST NUMBER
2463  014502  104006                            HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2464  014504  012767  000010  001144            MOV     #10,ICOUNT
2465  014512  004767  000676                    JSR     %7,RWALL        ;INITIALIZE ALL PAGES 4K, RW, BANK 0
2466  014516  004767  001746                    JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2467  014522  012777  077406  164346            MOV     #77406,@KPDR1   ;MAP KERNEL PAGE 1 RRW
2468  014530  012777  000001  164266            MOV     #1,@SR0         ;TURN ON SEGMENTATION
2469  014536  013737  020000  020000            MOV     @#20000,@#20000 ;READ AND WRITE PAGE 1
2470  014544  022777  077506  164324            CMP     #77506,@KPDR1   ;CHECK THE PDR
2471  014552  001401                            BEQ     .+4
2472  014554  104006                            HLT                             ;KERNEL PDR1 INCORRECT
2473                                                                            ;W BIT SHOULD BE SET
2474  014556  012701  020000                    MOV     #20000,R1
2475  014562  012702  000100                    MOV     #100,R2
2476  014566  005721                    L40:    TST     (R1)+           ;READ PAGE 1 REPEATEDLY
2477  014570  077202                            SOB     R2,L40
2478  014572  022777  077506  164276            CMP     #77506,@KPDR1   ;CHECK W BIT AGAIN
2479  014600  001401                            BEQ     .+4
2480  014602  104006                            HLT                             ;KERNEL PDR 1
2481                                                                            ;INCORRECT AFTER REPEATEDLY READING PAGE 1
2482  014604  005077  164214                    CLR     @SR0            ;TURN OFF SEGMENTATION
2483
2484                                    ;SHOW THAT IF MEMORY MANAG. IS ON, SETTING THE CURRENT MODE TO 10 WILL
2485                                    ;CAUSE A MEMORY MANAGEMENT ABORT.  NON RESIDENT SHOULD BE SET, AND ALSO PL SHOULD
2486                                    ;BE SET
2487  014610  104400                    TEST60: SCOPE
2488  014612  012706  001000                    MOV     #KSTACK,SP              ;INITIALIZE KERNEL STACK POINTER
2489  014616  004767  001562                    JSR     PC,ORDER                ;CHECK TEST SEQUENCE + INIT SR0
2490  014622  000060                            60                              ;TEST NUMBER
2491  014624  104006                            HLT                             ;TEST EXECUTED OUT OF SEQUENCE
2492  014626  012767  002000  001022            MOV     #2000,ICOUNT    ;RESTORE ITERATION COUNT
2493  014634  004767  000554                    JSR     %7,RWALL
2494  014640  004767  001624                    JSR     PC,KERN7                ;MAP KERNEL PAR/PDR 7 TO EXT BANK
2495  014644  012777  014712  164312            MOV     #RET42,@KTVEC   ;SETUP MEMORY MANAGEMENT ABORT RETURN
2496  014652  005077  164310                    CLR     @KTSTA
2497  014656  012777  000001  164140            MOV     #1,@SR0         ;TURN ON MEMORY MANAG.
2498  014664  012737  040000  177776    ADD42:  MOV     #40000,@#PS     ;SET MODE TO 01-FETCH OF NEXT
2499  014672  000240                            NOP                             ;INSTRUCTION SHOULD ABORT
2500  014674  005077  163076                    CLR     @PS             ;RESTORE MODE TO KERNEL
2501  014700  042777  000001  164116            BIC     #1,@SR0         ;TURN OFF MEMORY MANAG.
2502  014706  104006                            HLT                             ;NO ABORT WHEN MODE WAS SET
2503  014710  000415                            BR      CONT42          ;TO 01 (ILLEGAL)
2504  014712  042777  000001  164104    RET42:  BIC     #1,@SR0         ;TURN OFF MEMORY MANAG. AFTER ABORT
2505  014720  022777  100040  164076            CMP     #100040,@SR0    ;CK SR0
2506  014726  001401                            BEQ     .+4
2507  014730  104006                            HLT                             ;SR0 INCORRECT AFTER MODE 01 ABORT
2508                                                                            ;NR AND MODE 01 SHOULD BE SET
2509  014732  022777  014664  164072            CMP     #ADD42,@SR2     ;CHECK SR2
2510  014740  001401                            BEQ     .+4
2511  014742  104006                            HLT                             ;SR2 INCORRECT - SHOULD CONTAIN
```

```
2512                                                                    ;ADDRESS OF THE INSTRUCTION
2513                                                                    ;IMMEDIATELY AFTER THE ONE SETTING
2514                                                                    ;THE MODE TO 01
2515  014744  005077  164054          CONT42: CLR     @SR0              ;REINITIALIZE SR0
2516  014750  016777  164212  164206          MOV     KTSTA,@KTVEC      ;RESTORE TRAP CATCHER
2517
2518                                  ;#THIS TEST USES KPAR'S 0
2519                                  ;#AND 2 TO REFERENCE KPAR1 AND UPAR1 RESPECTIVELY.  A COUNT PATTERN IS
2520                                  ;#RUN THROUGH THE VIRTUAL ADDRESS STARTING AT BIT6 AND THE RECIPROCAL
2521                                  ;#COUNT PATTERN IS SIMULTANEOUSLY RUN THROUGH THE PAR'S.  AFTER A
2522                                  ;#RELOCATED REFERENCE IS MADE THE KT-11 IS TURNED OFF AND THE DATA IS
2523                                  ;#CHECKED TO ENSURE THAT, WHATEVER THE CONDITION OF THE BITS IS IN THE
2524                                  ;#VIRTUAL ADDRESS, THE DECODING FOR USER AND KERNAL PAR'S IS DONE BY
2525                                  ;#THE PHYSICAL ADDRESS.
2526  014756  104400          TEST61: SCOPE
2527  014760  012706  001000          MOV     #KSTACK,SP                ;INITIALIZE KERNEL STACK POINTER
2528  014764  004767  001414          JSR     PC,ORDER                  ;CHECK TEST SEQUENCE + INIT SR0
2529  014770  000061                  61                                ;TEST NUMBER
2530  014772  104006                  HLT                               ;TEST EXECUTED OUT OF SEQUENCE
2531  014774  004767  000414          JSR     %7,RWALL          ;SETUP ALL PDR'S FOR 4K R/W
2532  015000  004767  001464          JSR     PC,KERN7          ;SET UP KERNAL 7 REGISTERS
2533  015004  012777  007723  164102  MOV     #7723,@KPAR0      ;LOAD KPAR0 WITH ADDR OF KPAR1
2534  015012  012777  007776  164100  MOV     #7776,@KPAR2      ;LOAD KPAR2 WITH ADDR OF UPAR1
2535  015020  005000                  CLR     R0                ;CLEAR COUNTER REGISTER
2536  015022  012701  000042          MOV     #42,R1            ;LOAD OFFSET & BIT TO SELECT KPAR0
2537  015026  012702  040042          MOV     #40042,R2         ;LOAD OFFSET & BIT TO SELECT KPAR2
2538  015032  052777  000400  163764  3$:     BIS     #400,@SR0         ;TURN ON MAINTENENCE MODE
2539  015040  012711  005252          MOV     #5252,(R1)        ;LOAD PATTERN IN KERNAL PAR1
2540  015044  005077  163754          CLR     @SR0              ;TURN OFF MAINTENENCE MODE
2541  015050  027727  164042  005252  CMP     @KPAR1,#5252      ;DID DATA GET STORED IN KPAR1?
2542  015056  001401                  BEQ     1$                ;BRANCH IF DATA STORED CORRECTLY
2543  015060  104006                  HLT                       ;A HALT HERE INDICATES THAT THE
2544                                                            ;RELOCATION TO KPAR1 WAS NOT
2545                                                            ;SUCCESSFUL R1 HAS VIRTUAL ADDR AND
2546                                                            ;KPAR0 HAS THE BASE.
2547  015062  005077  164030          1$:     CLR     @KPAR1            ;CLEAR KPAR1 FOR NEXT TEST
2548  015066  052777  000400  163730          BIS     #400,@SR0         ;TURN ON MAINTENENCE MODE
2549  015074  012712  005252          MOV     #5252,(R2)        ;LOAD PATTERN IN USER PAR1
2550  015100  005077  163720          CLR     @SR0              ;TURN OFF MAINTENENCE MODE
2551  015104  027727  163746  005252  CMP     @UPAR1,#5252      ;DID DATA GET STORED IN UPAR1?
2552  015112  001401                  BEQ     2$                ;BRANCH IF DATA STORED CORRECTLY
2553  015114  104006                  HLT                       ;A HALT HERE INDICATES THAT THE
2554                                                            ;RELOCATION TO UPAR1 DID NOT WORK
2555                                                            ;R2 HAS THE VIRTUAL ADDR AND KPAR2
2556                                                            ;HAS THE BASE
2557  015116  005077  163734          2$:     CLR     @UPAR1            ;CLEAR UPAR1 FOR NEXT TEST
2558  015122  022700  010000          CMP     #10000,R0         ;CHECK TO SEE IF TEST IS DONE
2559  015126  001415                  BEQ     EOP               ;BRANCH IF TEST IS OVER
2560  015130  062700  000100          ADD     #100,R0           ;ADD BIT6 TO COUNTER
2561  015134  062701  000100          ADD     #100,R1           ;ADD BIT6 TO KPAR1'S VIRTUAL ADDR
2562  015140  062702  000100          ADD     #100,R2           ;ADD BIT6 TO UPAR1'S VIRTUAL ADDR
2563  015144  162777  000001  163742  SUB     #1,@KPAR0         ;SUBTRACT BIT1 FROM KPAR1'S BASE
2564  015152  162777  000001  163740  SUB     #1,@KPAR2         ;SUBTRACT BIT1 FROM UPAR1'S BASE
2565  015160  000724                  BR      3$                ;CONTINUE TEST
2566
2567
```

```
2568  015162  104400                    EOP:    SCOPE
2569  015164  032777  010000  164006            BIT     #BIT12,@SR              ;INHIBIT BELL?
2570  015172  001003                            BNE     1$              ;BRANCH IF BELL IS INHIBITED
2571  015174  012700  015360                    MOV     #BELL,R0        ;PUT ADDRESS OF BELL CHARS IN R0
2572  015200  000402                            BR      2$              ;BRANCH TO OUTPUT CODE
2573  015202  012700  015364            1$:     MOV     #ASTER,R0       ;PUT ADDRESS OF ASTERICK CHARS IN R0
2574  015206  112001                    2$:     MOVB    (R0)+,R1        ;CHECK FOR TERMINATOR CODE
2575  015210  001405                            BEQ     LOGICT          ;BRANCH IF BYTE IS ZERO
2576  015212  010177  163602            3$:     MOV     R1,@TDBR        ;OUTPUT CHARACTER TO BUFFER
2577  015216  105777  163574                    TSTB    @TCSR           ;SEE IF STATUS REG GETS SET
2578  015222  100373                            BPL     3$              ;BRANCH UNTIL IT DOES
2579  015224  013701  000042            LOGICT: MOV     @#42,R1         ;MONITOR HOOK
2580  015230  001405                            BEQ     END
2581  015232  000005                            RESET
2582  015234  004711                    LOGIC:  JSR     PC,@R1
2583  015236  000240                            NOP
2584  015240  000240                            NOP
2585  015242  000240                            NOP
2586  015244  000167  163734            END:    JMP     START
2587
2588                                    ;
2589                                    ;MESSAGE AREA
2590  015250  005015  030461  031457    MTIT:   .ASCII  <15><12>'11/34 MEMORY MANAGEMENT LOGIC TEST MAINDEC-11-DFKTA-A'<15><12>'
2591  015256  020064  042515  047515
2592  015264  054522  046440  047101
2593  015272  043501  046505  047105
2594  015300  020124  047514  044507
2595  015306  020103  042524  052123
2596  015314  046440  044501  042116
2597  015322  041505  030455  026461
2598  015330  043104  052113  026501
2599  015336  006501  040012
2600  015342  005015  041520  020075  MPC:    .ASCII  <15><12>'PC= @'
2601  015350      100
2602  015351      040  050040  036523  MPS:    .ASCII  '  PS= @'
2603  015356  040040
2604  015360  177607  000377            BELL:   .ASCIZ  <207><377><377>
2605  015364  177452  000377            ASTER:  .ASCIZ  /*/<377><377>
2606                                            .EVEN
2607
2608
2609                                    ;SUBROUTINE TO CLEAR ALL MEMORY MANAG. REGISTERS (EXCEPT SR1,SR2)
2610  015370  005077  163430            CLRALL: CLR     @SR0
2611  015374  005000                            CLR     R0
2612  015376  012701  000040                    MOV     #32.,R1                 ;COUNT OF REGISTERS TO BE CLEARED
2613  015402  005070  001034            CLRLP:  CLR     @ADRTAB(R0)             ;CLEAR REGISTERS THRU ADDRESS TABLE
2614  015406  005720                            TST     (R0)+                   ;MOVE POINTER
2615  015410  077104                            SOB     R1,CLRLP                ;LOOP TILL DONE
2616  015412  000207                            RTS     %7
2617
2618                                    ;SUBROUTINE TO MAKE ALL PAGES RW, BANK 0, 4K, UP
2619  015414  005077  163404            RWALL:  CLR     @SR0
2620  015420  012701  001034                    MOV     #ADRTAB,R1      ;R1 POINTS TO ADDRESS TABLE
2621  015424  012700  000010            RWL1:   MOV     #10,R0          ;R0 IS COUNTER
2622  015430  005071  000020            RWL2:   CLR     @20(R1)         ;CLEAR PAR
2623  015434  012731  077406                    MOV     #77406,@(R1)+   ;SET PDR RW, 4K
```

```
2624  015440  077005                        SOB     R0,RWL2
2625  015442  062701  000020                ADD     #20,R1
2626  015446  020127  001132                CMP     R1,#ADREND      ;POINTER TO NEXT GROUP
2627  015452  002764                        BLT     RWL1
2628  015454  000207                        RTS     %7
2629
2630                                  ;ROUTINE TO LOOP THRU A SINGLE INSTRUCTION TEST
2631                                  ;LOAD THE STARTING ADDRESS OF THE TEST
2632                                  ;YOU WISH TO RUN (THE ADDRESS OF THE TESTX
2633                                  ;TAG) IN LOCATION "RETRNX" BELOW
2634                                  ;NOTE THAT SW11 MUST BE DOWN TO RUN THIS TEST.
2635  015456  005037  177776          TESTX:  CLR     @#PS
2636  015462  012706  001000                  MOV     #KSTACK,SP
2637  015466  012737  140000  177776        MOV     #140000,@#PS            ;SETUP USER TRAP
2638  015474  012706  000400                MOV     #USTACK,SP
2639  015500  005037  177776                CLR     @#PS
2640  015504  062767  000002  000030        ADD     #2,RETRNX               ;ADD 2 TO POINT TO INSTRUCTION AFTER
2641  015512  000000                        HALT                            ;SET SR OPTIONS
2642  015514  005067  000140                CLR     SCOPEF                  ;KEEP COUNT AT ZERO
2643  015520  012767  015532  000134        MOV     #XLOOP,RETURN           ;LOAD SCOPE LOOP RETURN POINTER
2644  015526  000177  000010                JMP     @RETRNX                 ;JUMP TO TEST
2645  015532  005067  000122          XLOOP:  CLR     SCOPEF                  ;KEEP COUNT AT ZERO
2646  015536  000177  000000                JMP     @RETRNX                 ;JUMP TO TEST
2647  015542  000000                  RETRNX: 0
2648                                  ;SCOPE AND/OR ITERATION LOOP FOR EACH TEST 4000 TIMES
2649  015544  032777  040000  163426  SCOPEC: BIT     #BIT14,@SR              ;TEST SR FOR SCOPE
2650  015552  001015                        BNE     SCOPEB                  ;YES,SCOPE
2651  015554  032777  004000  163416        BIT     #BIT11,@SR              ;NO-TEST FOR ITERATION
2652  015562  001020                        BNE     SCOPEG                  ;INHIBIT ITERATION
2653  015564  026767  000070  000064        CMP     SCOPEF, ICOUNT          ;COMPARE CURRENT COUNT TO MAX NUMBER
2654  015572  100014                        BPL     SCOPEG                  ;EXIT-DONE
2655  015574  005267  000060                INC     SCOPEF                  ;INCREMENT COUNT
2656  015600  012737  000340  177776        MOV     #340,@#PS               ;PREVENT TRAPPING WHILE MOVING STACK
2657  015606  022626                  SCOPEB: CMP     (6)+,(6)+                       ;REPOSITION STACK
2658  015610  005037  177776                CLR     @#PS
2659  015614  005077  163204                CLR     @SR0
2660  015620  000177  000036                JMP     @RETURN                 ;REPEAT TEST
2661  015624  005067  000030          SCOPEG: CLR     SCOPEF                  ;CLEAR COUNT
2662  015630  005267  163340                INC     TESTCT                  ;STEP TEST COUNTER TO ALLOW CHECKING
2663                                                                          ;ORDER OF EXECUTION.
2664  015634  011667  000022                MOV     @%6,RETURN              ;SAVE SCOPE RETURN POINTER
2665  015640  022626                        CMP     (6)+,(6)+                       ;RETURN INLINE-NEXT TEST
2666  015642  005037  177776                CLR     @#PS
2667  015646  005077  163152                CLR     @SR0
2668  015652  000177  000004                JMP     @RETURN
2669  015656  004000                  ICOUNT: 4000                            ;ITERATION COUNT
2670  015660  000000                  SCOPEF: 0                               ;COUNT LOCATION FOR ITERATION LOOP
2671  015662  000000                  RETURN: 0                               ;ADDRESS OF LAST TEST
2672
2673                                  ;ENTERED WITH SYSTEM TRAP CALL (HLT)
2674                                  ;PRINT OUT THE ERROR PC+2 AND STATUS REGISTER
2675  015664  012767  000340  162104  PRINT:  MOV     #340,PS                 ;SET PRIORITY TO 7
2676  015672  032777  020000  163300        BIT     #BIT13,@SR              ;TEST FOR INHIBIT PRINT OUT
2677  015700  001401                        BEQ     .+4                     ;BRANCH TO PRINT
2678  015702  000430                        BR      CK                      ;INHIBIT, CHECK FOR HALT
2679  015704  012667  000066                MOV     (6)+,SAVPC              ;PC OF FAILING ROUTINE
```

```
 2680  015710  012667  000064                MOV     (6)+,SAVPSR             ;PSR OF ERROR CONDITION
 2681  015714  024646                        CMP     -(6),-(6)               ;RESTORE STACK
 2682  015716  012767  000200  162052        MOV     #200,PS
 2683  015724  016767  000046  000374        MOV     SAVPC,PTEMP1            ;LOAD WITH FAILING PC+2
```

```
2684  015732  004767  000044                        JSR     PC,TYPE
2685  015736  015342                                MPC
2686  015740  004767  000116                        JSR     PC,PRSHRT               ;PRINT FAILING PC+2
2687  015744  004767  000032                        JSR     PC,TYPE
2688  015750  015351                                MPS
2689  015752  016767  000022  000346                MOV     SAVPSR,PTEMP1           ;LOAD PROCESSOR STATUS
2690  015760  004767  000130                        JSR     PC,PROCT                ;PRINT PROCESSOR STATUS
2691  015764  005777  163210            CK:         TST     @SR                     ;CHECK SR FOR HALT SWITCH
2692  015770  100001                                BPL     .+4                     ;BRANCH IF NOT SET
2693  015772  000000                                HALT                            ;HALT ON ERROR UP
2694  015774  000002                                RTI                             ;RETURN TO MAIN LINE
2695  015776  000000            SAVPC:      0
2696  016000  000000            SAVPSR:     0
2697
2698                                    ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE
2699  016002  010067  000052            TYPE:       MOV     %0,SAVR0
2700  016006  011600                                MOV     (6),%0                  ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
2701  016010  062716  000002                        ADD     #2,@%6                  ;SET UP EXIT
2702  016014  011000                                MOV     @%0,%0
2703  016016  112067  000034            TYPA:       MOVB    (0)+,TYPDAT             ;GET CHARACTER
2704  016022  122767  000100  000026                CMPB    #100,TYPDAT             ;CHECK FOR "@" CHARACTER
2705  016030  001003                                BNE     TYPB                    ;BRANCH IF NOT "@"
2706  016032  016700  000022                        MOV     SAVR0,%0                ;RESTORE R0
2707  016036  000207                                RTS     PC                      ;TERMINATOR CHAR. EXIT
2708  016040  116777  000012  162752  TYPB:       MOVB    TYPDAT,@TDBR            ;OUTPUT CHAR TO PRINTER
2709  016046  105777  162744                        TSTB    @TCSR                   ;WAIT FOR TTY READY
2710  016052  100375                                BPL     .-4
2711  016054  000760                                BR      TYPA
2712  016056  000000            TYPDAT:     0
2713  016060  000000            SAVR0:      0
2714
2715                                    ;SUBROUTINE TO PRINT OUT OCTAL NUMBER
2716                                    ;PRSHRT DELETES LEADING ZEROS
2717                                    ;PROCT PRINTS OUT 6 OCTAL DIGITS
2718  016062  012767  000001  000232  PRSHRT:     MOV     #1,PRSFLG               ;SET FLAG TO INDICATE SHORT PRINTOUT
2719  016070  005767  000232                        TST     PTEMP1                  ;CHECK FOR ZERO
2720  016074  001011                                BNE     PROCT+44                ;BRANCH IF NOT ZERO
2721  016076  012777  000260  162714                MOV     #260,@TDBR              ;OUTPUT A SINGLE ZERO
2722  016104  105777  162706                        TSTB    @TCSR                   ;WAIT FOR TTY READY
2723  016110  100375                                BPL     .-4
2724  016112  000207                                RTS     %7                      ;RETURN
2725  016114  005067  000202            PROCT:      CLR     PRSFLG                  ;CLEAR FLAG TO INDICATE FULL PRINTOUT
2726  016120  005067  000206                        CLR     PTEMP3                  ;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
2727  016124  005067  000174                        CLR     PRFLG                   ;INITIALIZE CARRY FLAG FOR ROTATES
2728  016130  012767  000260  000172                MOV     #260,PTEMP2             ;SETUP R3
2729  016136  005767  000164                        TST     PTEMP1                  ;CHECK BIT 15 OF NUMBER
2730  016142  100002                                BPL     .+6                     ;BRANCH IF ZERO
2731  016144  005267  000160                        INC     PTEMP2                  ;INCREMENT R3 IF ONE
2732  016150  006167  000152                        ROL     PTEMP1                  ;ROTATE LEFT MOST OCTAL TO RIGHT END
2733  016154  006167  000146                        ROL     PTEMP1
2734  016160  005567  000140                        ADC     PRFLG                   ;STORE CARRY
2735  016164  005767  000132            P.CK:       TST     PRSFLG                  ;CHECK FOR SHORT PRINTOUT
2736  016170  001404                                BEQ     P.WAIT                  ;BRANCH IF NOT SET
2737  016172  026727  000132  000260                CMP     PTEMP2,#260             ;CHECK FOR ZERO IF SET
2738  016200  001410                                BEQ     P.CONT                  ;IF SET, GO TO NEXT CHARACTER
2739  016202  016777  000122  162610  P.WAIT:      MOV     PTEMP2,@TDBR            ;OUTPUT NEXT CHARACTER
```

D05

```
2740  016210  105777  162602                    TSTB    @TCSR           ;WAIT FOR TTY READY
2741  016214  100375                            BPL     .-4
2742  016216  005067  000100                    CLR     PRSFLG          ;PRINT REST OF NUMBER AFTER A NON-ZERO DIGIT
2743  016222  005267  000104          P.CONT:   INC     PTEMP3          ;COUNT
2744  016226  026727  000100  000006            CMP     PTEMP3,#6       ;CHECK FOR DONE
2745  016234  001001                            BNE     P.CNT1          ;BRANCH IF NOT DONE
2746  016236  000207                            RTS     %7
2747  016240  000241          P.CNT1:   CLC                             ;CLEAR CARRY
2748  016242  005767  000056                    TST     PRFLG           ;CHECK FOR PREVIOUS CARRY
2749  016246  001403                            BEQ     .+10            ;BRANCH IF PREVIOUSLY ZERO
2750  016250  005067  000050                    CLR     PRFLG           ;INITIALIZE FLAG
2751  016254  000261                            SEC                     ;SET CARRY
2752  016256  006167  000044                    ROL     PTEMP1          ;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTE
2753  016262  006167  000040                    ROL     PTEMP1
2754  016266  006167  000034                    ROL     PTEMP1
2755  016272  005567  000026                    ADC     PRFLG           ;STORE CARRY
2756  016276  016767  000024  000024            MOV     PTEMP1,PTEMP2   ;LOAD DATA INTO R3
2757  016304  042767  177770  000016            BIC     #177770,PTEMP2  ;CLEAR ALL BUT LOWEST OCTAL DIGIT
2758  016312  052767  000260  000010            BIS     #260,PTEMP2     ;SET TO ASCII EQUIVALENT
2759  016320  000721                            BR      P.CK            ;LOOP
2760  016322  000000          PRSFLG:   0
2761  016324  000000          PRFLG:    0
2762  016326  000000          PTEMP1:   0                               ;CONTAINS VALUE TO BE OUTPUT
2763  016330  000000          PTEMP2:   0                               ;SCRATCH
2764  016332  000000          PTEMP3:   0                               ;USED TO COUNT CHARACTERS OUTPUT
2765
2766                          ;EMT HANDLER
2767                          ;FIRST 3 CALLS LEFT OPEN IN TABLE FOR EASY PATCHES
2768  016334  011667  000032  EMTSRV:   MOV     @SP,EPC                 ;GET CALL
2769  016340  162767  000002  000024            SUB     #2,EPC
2770  016346  017767  000020  000016            MOV     @EPC,EPC
2771  016354  105067  000013                    CLRB    EPC+1           ;SAVE OFFSET ONLY
2772  016360  062767  016374  000004            ADD     #EMTAB,EPC      ;POINT TO TABLE OF ADDRESSES
2773  016366  017707  000000                    MOV     @EPC,PC         ;JUMP TO DESIRED ROUTINE
2774  016372  000000          EPC:      0
2775          104000                    PATCH1=EMT+0
2776          104002                    PATCH2=EMT+2
2777          104004                    PATCH3=EMT+4
2778  016374  104000          EMTAB:    PATCH1
2779  016376  104002                    PATCH2
2780  016400  104004                    PATCH3
2781  016402  015664                    PRINT
2782                          ;
2783                          ;SUBROUTINE TO CHECK TEST SEQUENCE
2784  016404  005037  177776  ORDER:    CLR     @#PS                    ;CLEAR PROCESSOR STATUS
2785  016410  011667  000052            MOV     (SP),TEMPN              ;GET TEST NUMBER ADDRESS
2786  016414  017767  000046  000044    MOV     @TEMPN,TEMPN            ;GET TEST NUMBER
2787  016422  032777  002000  162550    BIT     #BIT10,@SR
2788  016430  001404                    BEQ     ORDERB
2789  016432  016700  000030            MOV     TEMPN,R0
2790  016436  000005                    RESET
2791  016440  000000                    HALT
2792  016442  026767  162526  000016  ORDERB:  CMP     TESTCT,TEMPN    ;IS TEST SEQUENCE CORRECT
2793  016450  001403                    BEQ     ORDERA                  ;YES, CONTINUE
2794  016452  062716  000002            ADD     #2,(SP)                 ;UPDATE FOR ERROR RETURN
2795  016456  000207                    RTS     PC
```

EOS


```
2796  016460  062716  000004          ORDERA: ADD     #4,(SP)                 ;UPDATE FOR GOOD RETURN
2797  016464  000207                          RTS     PC
2798  016466  000000                  TEMPN:  0
2799                                  ;MAP KERNEL PAR/PDR 7 TO EXTERNAL BANK
2800  016470  012777  007600  162434  KERN7:  MOV     #7600,@KPAR7
2801  016476  012777  077406  162406          MOV     #77406,@KPDR7
2802  016504  000207                          RTS     PC
2803          017712                          .=17712
2804
2805  017712  125252                  DESTAD: 125252
2806          000001                          .END
```

F05


```
AOD23   004726          1104#   1117
AOD24   005126          1143#   1154
AOD42   014664          2498#   2509
AOREND= 001132           382#    567     1238    1263    2404    2421    2626
AORTAB  001034           346#    544      546    1230    1251    2391    2410    2613*   2620
AOR11   012136          2010#   2020
AOR2    011076          1818    1824#
AOR25   005322          1182#   1192    1197
AOR25A  005324          1183#   1207
AOR36   007560          1600#   1620
AOR36A  007642          1613#   1623
AOR36B  007740          1632#   1640
AOR36C  007742          1633#   1644
AOR37   010114          1668#   1688
AOR37A  010176          1681#   1691
AOR37B  010274          1700#   1708
AOR37C  010276          1701#   1712
AOR40   010450          1736#   1756
AOR40A  010532          1749#   1759
AOR40B  010630          1768#   1776
AOR40C  010632          1769#   1780
AO13    002572           720#    721
AO13A   002606           724#    726
AO13B   002636           730#    732
AO13C   002666           736#    738
AO21A = 004020           930     931#
AO21B = 004032           936     937#
AO21C   004142           943*    965#
ASTER   015364          2573    2605#
BELL    015360          2571    2604#
BIT0  = 000001           286#
BIT1  = 000002           287#
BIT10 = 002000           296#   2787
BIT11 = 004000           297#   2651
BIT12 = 010000           298#   2569
BIT13 = 020000           299#    736    2676
BIT14 = 040000           300#    730     735    2649
BIT15 = 100000           301#    724     729
BIT2  = 000004           288#
BIT3  = 000010           289#
BIT4  = 000020           290#
BIT5  = 000040           291#
BIT6  = 000100           292#
BIT7  = 000200           293#
BIT8  = 000400           294#
BIT9  = 001000           295#
BLK20   003546           880     883#    895
BLOCKS  001176           404#    876*    894*
CK      015764          2678    2691#
CKWBIT  013524          2278    2285#
CLRALL  015370           484     515     543     583     610     751     778     804     823     853     918     981    1098
                        1138    1175    1308    1593    1661    1729    2250    2610#
CLRLP   015402          2613#   2615
CNT22B  004276           999    1007#
CNT22C  004356          1018    1026#
CNT22D  004452          1038    1046#
```

```
CNT22E  004546            1060    1069#
CNT31A  005570            1239    1241#
CNT31B  005660            1254    1260#
CONT11  012202            2015    2024#
CONT23  012664            2110    2115    2127#
CONT24  013100            2166    2171    2182#
CONT42  014744            2503    2515#
CONT5   001760             553     557     563#
C25     013254            2222#   2237
DATA16  003112             781     787     788     796#    809     826
DESTAD  017712            1008*   1102*   1118    2805#
DISPLA  001202             406#    419*
DISPRE  000174             319#    419
DONE13  012346            2044    2054#
DONE16  012460            2074    2077#
DONE2   011126            1825    1835#
DONE23  012676            2105    2132#
DONE24  013112            2161    2187#
DONE25  013324            2227    2240#
DONE36  010016            1603    1615    1635    1648#
DONE37  010352            1671    1683    1703    1716#
DONE40  010706            1739    1751    1771    1784#
DONE5   011560            1912    1926#
DONE5A  001776             569     574#
DONE7   011756            1946    1958    1965#
DON25   005440            1185    1211#
DST21A  004134             923     926*    942*    946     960*    962#
DST21B  004136             924     927*    950     959*    963#    965
DST21C  004140             925     928*    944*    954     958*    964#
EMTAB   016374            2772    2778#
EMTSRV  016334             310    1414    1419    2768#
END     015244            2580    2586#
END20   003632             898     901#
END22E  004636            1083    1087#
END35   007464            1574    1580#
EOP     015162            2559    2568#
EPC     016372            2768*   2769*   2770*   2771*   2772*   2773    2774#
ERR22A  004264             993    1000#
ERR22B  004350            1012    1019#
ERR22C  004436            1032    1039#
ERR22D  004532            1053    1061#
ERR22E  004630            1076    1084#
EXIT10  002360             660     665#
EXIT11  002510             695     700#
EXIT2   001474             458     472#
EXIT21  004144             961     966#
EXIT23  005046            1116    1128#
EXIT24  005242            1152    1161#
EXT27   013560            2284    2296#
FTITLE  001172             402#    426     430*
HLT   = 104006             285#    438     442     445     454     457     467     482     493     514     526     542     558
                           581     592     598     609     619     625     636     654     672     689     707     712     719
                           723     728     734     740     750     776     803     819     852     916     933     939     948
                           952     956     980     998    1001    1017    1020    1037    1040    1059    1062    1081    1085
                          1096    1113    1136    1150    1174    1190    1194    1199    1202    1206    1209    1222    1234
                          1250    1257    1268    1284    1295    1307    1334    1337    1340    1343    1362    1415    1464
```

```
                    1718*  1733*  1744*  1763*  1786*  1815*  1836*  1851*  1868*  1900*  1926*  1941*  1952*
                    1965*  2005*  2025*  2037*  2054*  2066*  2078*  2095*  2102*  2133*  2152*  2159*  2188*
                    2202*  2240*  2495*  2516*
K123    001012       334#   859
K134    001014       335#   860
LOGIC   015234       316   2582#
LOGICT  015224      2575   2579#
LOOP10  002266       642#   646
LOOP11  002416       677#   681
LOOP2   001434       460#   470
LOOP23  012542      2097#  2131
LOOP24  012764      2154#  2186
LOOP3   001534       487#   502
LOOP3A  001542       489#   501
LOOP3B  001546       490#   498
LOOP32  014164      2381#  2390
LOOP4   001624       517#   535
LOOP4A  001632       520#   533
LOOP4B  001636       521#   530
LOOP5   011464      1906#  1925
LOOP6   002044       586#   602
LOOP6A  002052       588#   601
LOOP7   002150       613#   629
LOOP7A  002156       615#   628
LOP10A  002274       644#   645
LOP10B  002314       650#   658
LOP10C  002322       652#   657
LOP11A  002424       679#   680
LOP11B  002444       685#   693
LOP11C  002452       687#   692
LOP27   013456      2270#  2283
LOP27A  013470      2275#  2280
LOP31A  005536      1231#  1240
LOP31B  005542      1232#  1236
LOP31C  005574      1242#  1275
LOP31D  005606      1245#  1273
LOP31E  005636      1252#  1264
LOP31F  005642      1253#  1261
LOP32C  014202      2385#  2387
LOP32D  014230      2392#  2405
LOP32E  014234      2394#  2401
LOP32F  014304      2411#  2422
LOP32G  014310      2413#  2418
LOPSA   011542      1919   1921#
LOPSAA  001722       546#   571
LOPSB   001740       552#   564
L25A    013210      2207#  2226
L25B    013220      2209#  2224
L40     014566      2476#  2477
MPC     015342      2600#  2685
MPS     015351      2602#  2688
MTIT    015250       429   2590#
NG35B   006564      1373   1412#
NG35C   006774      1423   1462#
NG35D   007216      1472   1517#
NG35E   007436      1530   1570#
```

```
NOP    = 000240   272#
ODDAD    007242  1516    1521    1525#
OK35B    006602  1374    1417#
OK35C    007004  1424    1466#
OK35D    007232  1473    1522#
OK35E    007446  1532    1575#
ORDER    016404   436     452     480     512     540     579     607     634     670     705     717     748     774
                  801     850     914     978    1094    1134    1172    1220    1282    1305    1360    1590    1658
                 1726    1791    1807    1844    1873    1895    1934    1971    1998    2031    2061    2086    2143
                 2196    2246    2261    2302    2335    2370    2435    2461    2489    2528    2784#
ORDERA   016460  2793    2796#
ORDERB   016442  2788    2792#
PAGES    001156   396#    871*    878     897*
PAG20    003510   876#    900
PARTAB   001140   386#    485     584     640     648
PATCH1=  104000  2775#    2778
PATCH2=  104002  2776#    2779
PATCH3=  104004  2777#    2780
PC     =X000007   282#    428*    436*    452*    480*    512*    540*    579*    607*    634*    670*    705*    717*
                  748*    774*    801*    810*    850*    914*    921*    978*    984*   1094*   1101*   1134*   1141*
                 1172*   1177*   1220*   1225*   1282*   1287*   1305*   1318*   1360*   1366*   1590*   1594*   1658*
                 1662*   1726*   1730*   1791*   1807*   1813*   1844*   1850*   1873*   1877*   1895*   1899*   1934*
                 1938*   1971*   1998*   2004*   2031*   2036*   2061*   2065*   2086*   2090*   2143*   2147*   2196*
                 2201*   2246*   2261*   2265*   2302*   2306*   2335*   2339*   2370*   2375*   2435*   2440*   2461*
                 2466*   2489*   2494*   2528*   2532*   2582*   2684*   2686*   2687*   2690*   2707*   2773*   2795*
                 2797*   2802*
PDREND   001136   385#    534
PDRM2    001170   401#    522
PDRTAB   001134   384#    516     611     675     683
PRFLG    016324  2727*   2734*   2748    2750*   2755*   2761#
PRINT    015664  2675#   2781
PROCT    016114  2690    2720    2725#
PRSFLG   016322  2718*   2725*   2735    2742*   2760#
PRSHRT   016062  2686    2718#
PS     = 177776   283#    284     410*    830*    862*    864*    869*    903*    989*    996    1003*   1227*   1229*
                 1244*   1310*   1312*   1327*   1369*   1371*   1377*   1427*   1476*   1529*   1534*   1577    1853*
                 1907*   1953*   1976*   1979*   2267*   2274*   2377*   2382*   2406*   2498*   2500*   2635*   2637*
                 2639*   2656*   2658*   2666*   2675*   2682*   2784#
PTEMP1   016326  2683*   2689*   2719    2729    2732*   2733*   2752*   2753*   2754*   2756    2762#
PTEMP2   016330  2728*   2731*   2737    2739    2756*   2757*   2758*   2763#
PTEMP3   016332  2726*   2743*   2744    2764#
P.CK     016164  2735#   2759
P.CNT1   016240  2745    2747#
P.CONT   016222  2738    2743#
P.WAIT   016202  2736    2739#
RETRNX   015542  2640*   2644    2646    2647#
RETURN   015662   424*   2643*   2660    2664*   2668    2671#
RET11    012150  2005    2016#
RET13A   012312  2037    2045#
RET16    012450  2066    2075#
RET2     011102  1815    1826#
RET23A   012610  2095    2111#
RET23B   012630  2102    2116#
RET24A   013032  2152    2167#
RET24B   013044  2159    2172#
RET25    013276  2202    2228#
```

```
SCOPEF  015660          2642* 2645* 2653  2655* 2661* 2670#
SCOPEG  015624          2652  2654  2661#
SP     =%000006          281#  411*  413*  414*  420   421   422   435*  451*  479*  511*  539*  578*
                         606*  633*  669*  704*  716*  747*  773*  800*  849*  863*  913*  923*  924*
                         925*  958   959   960   977*  1070* 1087  1093* 1133* 1171* 1219* 1281* 1304*
                        1309* 1311* 1350* 1359* 1370* 1372* 1412  1417  1462  1466  1517  1522  1570
                        1575  1589* 1644  1657* 1712  1725* 1780  1790* 1806* 1817* 1818* 1843* 1872*
                        1894* 1933* 1970* 1997* 2007* 2008* 2030* 2045  2060* 2085* 2112  2116  2142*
                        2169  2172  2195* 2228  2245* 2260* 2301* 2334* 2369* 2434* 2460* 2488* 2527*
                        2636* 2638* 2768  2785  2794* 2796*
SR      001200           405#  416   418*  2569  2649  2651  2676  2691  2787
SR0     001024           340#  440   455   464*  465   471*  724*  729*  730*  735*  736*  757*  762
                         768*  782*  789*  811   812*  817   829   831*  873   883*  904*  922   929*
                         990*  995*  1000* 1010* 1014* 1019* 1029* 1034* 1039* 1049* 1055* 1061* 1073*
                        1078* 1084* 1103* 1110  1115* 1142* 1148  1155* 1181* 1184* 1186  1187* 1245*
                        1247* 1276* 1290* 1293  1297* 1317  1321* 1328* 1363* 1368  1378* 1428* 1477*
                        1535* 1599* 1601* 1604* 1605  1611* 1613* 1616* 1617  1628* 1631* 1633* 1636*
                        1637  1648* 1667* 1669* 1672* 1673  1679* 1681* 1684* 1685  1696* 1699* 1701*
                        1704* 1705  1716* 1735* 1737* 1740* 1741  1747* 1749* 1752* 1753  1764* 1767*
                        1769* 1772* 1773  1784* 1819* 1826  1832  1835* 1855* 1858  1863  1866* 1878*
                        1879* 1880  1883* 1885  1888* 1908* 1913  1914* 1942* 1944* 1947  1948* 1954*
                        1956* 1959  1960* 1977* 1978* 1980  1988* 2010* 2016* 2017  2024* 2039* 2040*
                        2042* 2046  2047* 2056* 2068* 2069  2070* 2075* 2077* 2097* 2108* 2111* 2114*
                        2117  2118* 2123  2132* 2154* 2164* 2167* 2173  2174* 2179  2187* 2206* 2241*
                        2275* 2277* 2307* 2309* 2318* 2320* 2341* 2343* 2352* 2354* 2384* 2388* 2407*
                        2423* 2442* 2455* 2468* 2482* 2497* 2501* 2504* 2505  2515* 2538* 2540* 2548*
                        2550* 2610* 2619* 2659* 2667*
SR0H    001026           341#  1794* 1795  1799  1884*
SR1     001030           342#  443   709*  710
SR2     001032           343#  720   726   732   738   1192  1196* 1197  1620  1640  1688  1708  1756
                        1776  2020  2509
STAEND  001152           392#  901   1274  2282  2389
STAPNT  001154           395#  865*  866   868*  869   870*  901
START   001204           324   410#  2586
STATAB  001144           389#  865   1241  2269  2379
STATUS= 177776           284#
STAT2D  003430           866#  902
SWREG   000176           320#  418
TCSR    001016           336#  1514* 1519* 1524* 2577  2709  2722  2740
TDBR    001020           337#  2576* 2708* 2721* 2739*
TEMP    001022           338#  2449* 2450
TEMPN   016466          2785* 2786* 2789  2792  2798#
TESTCT  001174           403#  425*  2662* 2792
TESTN = 000062           408#  434#  450#  478#  510#  538#  577#  605#  632#  668#  703#  715#  746#
                         772#  799#  848#  912#  976#  1092# 1132# 1170# 1218# 1280# 1303# 1358# 1588#
                        1656# 1724# 1789# 1805# 1842# 1871# 1893# 1932# 1969# 1996# 2029# 2059# 2084#
                        2141# 2194# 2244# 2259# 2300# 2333# 2368# 2433# 2459# 2487# 2526#
TESTX   015456           326   2635#
TEST1   001332           424   427   434#
TEST10  002222           632#
TEST11  002360           668#
TEST12  002512           703#
TEST13  002554           715#
TEST14  002710           746#
TEST15  002774           772#
TEST16  003114           799#
```

```
TEST17  003310            848#
TEST2   001400            450#
TEST20  003700            912#
TEST21  004146            976#
TEST22  004642           1092#
TEST23  005050           1132#
TEST24  005244           1170#
TEST25  005452           1218#
TEST26  005740           1280#
TEST27  006044           1303#
TEST3   001474            478#
TEST30  006330           1358#
TEST31  007500           1588#
TEST32  010034           1656#
TEST33  010370           1724#
TEST34  010724           1789#
TEST35  011004           1805#
TEST36  011146           1842#
TEST37  011304           1871#
TEST4   001576            510#
TEST40  011406           1893#
TEST41  011572           1932#
TEST42  011764           1969#
TEST43  012056           1996#
TEST44  012214           2029#
TEST45  012360           2059#
TEST46  012472           2084#
TEST47  012714           2141#
TEST5   001674            538#
TEST50  013130           2194#
TEST51  013336           2244#
TEST52  013404           2259#
TEST53  013560           2300#
TEST54  013732           2333#
TEST55  014104           2368#
TEST56  014344           2433#
TEST57  014466           2459#
TEST6   002004            577#
TEST60  014610           2487#
TEST61  014756           2526#
TEST7   002116            605#
TST10   002256            640#    663
TST10F  002356            638*    659     661*    664#
TST11   002406            675#    698
TST11F  002506            673*    694     696*    699#
TYPA    016016           2703#   2711
TYPB    016040           2705    2708#
TYPDAT  016056           2703*   2704    2708    2712#
TYPE    016002            428    2684    2687    2699#
UPAR0   001054            356#     824*  1365*
UPAR1   001056            357#   2551    2557*
UPAR2   001060            358#
UPAR3   001062            359#
UPAR4   001064            360#
UPAR5   001066            361#
UPAR6   001070            362#
```

N05


```
UPAR7   001072          363#   827*  1226*  1320*  1367*  1975*  2266*  2376*
UPDR0   001034          347#   391    825*  1314*  1902   1918
UPDR1   001036          348#  1940*
UPDR2   001040          349#
UPDR3   001042          350#
UPDR4   001044          351#
UPDR5   001046          352#
UPDR6   001050          353#
UPDR7   001052          354#   828*  1319*  1849*
URET34  006226         1325   1331#  1338
USTACK  000400          329#   863   1228   1370   1854   2268   2378   2638
XLOOP   015532         2643   2645#
.     = 017714          307#   309#   312#   315#   318#   323#   325#   328#   330#   382    431    441    444
                        456    466    492    525    591    597    618    624    653    688    711    722    727
                        733    739    763    784    791    795    815    818    834    835    885    888    931
                        932    937    938    947    951    955    997   1016   1036   1058   1080   1112   1117#
                       1121   1127   1149   1154#  1160   1189   1193   1198   1201   1205   1208   1233   1249
                       1256   1267   1294   1333   1336   1339   1342   1578   1606   1618   1621   1624   1638
                       1641   1645   1674   1686   1689   1692   1706   1709   1713   1742   1754   1757   1760
                       1774   1777   1781   1796   1800   1827   1833   1857   1859   1864   1881   1886   1910
                       1916   1950   1962   1981   2008   2018   2021   2049   2052   2072   2120   2124   2176
                       2180   2219   2235   2253   2286   2292   2311   2315   2322   2328   2345   2349   2356
                       2362   2395   2414   2445   2453   2471   2479   2506   2510   2677   2692   2710   2723
                       2730   2741   2749   2803#
```

ABRT	1583#	1584	1652	1720											
TESTNO	408#	434	450	478	510	538	577	605	632	668	703	715	746	772	799
	848	912	976	1092	1132	1170	1218	1280	1303	1358	1588	1656	1724	1789	1805
	1842	1871	1893	1932	1969	1996	2029	2059	2084	2141	2194	2244	2259	2300	2333
	2368	2433	2459	2487	2526										

DFKTAA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
ENT     1411  2775  2776  2777
HALT     307   764   765   792   816   886   889  1104  1105  1106  1107  1108  1109  1122  1126
        1143  1144  1145  1158  1183  1242  1824  2014  2641  2643  2791
INC      430   661   696  1181  1245  1290  1321  1328  1378  1428  1477  1535  1599  1631  1667
        1698  1735  1767  1819  1865  1908  1942  1954  1977  2039  2068  2097  2154  2275  2307
        2318  2325  2341  2352  2359  2384  2407  2655  2662  2731  2743
IOT     1322  1329  1461
JMP      324   326  2586  2644  2646  2660  2668
JSR      428   436   452   480   484   512   515   540   543   578   583   607   610   634   670
         705   717   748   751   774   778   801   804   810   823   850   853   854   914   918
         921   978   981   984  1094  1098  1101  1134  1138  1141  1172  1175  1177  1220  1224
        1235  1282  1285  1287  1305  1308  1318  1360  1364  1366  1590  1593  1594  1658  1661
        1662  1726  1729  1730  1791  1807  1810  1813  1844  1847  1850  1873  1876  1877  1895
        1898  1899  1934  1937  1938  1971  1974  1998  2001  2004  2031  2034  2036  2061  2064
        2065  2086  2089  2090  2143  2146  2147  2196  2200  2201  2246  2250  2261  2264  2265
        2278  2302  2305  2306  2336  2338  2339  2370  2374  2375  2435  2439  2440  2461  2465
        2466  2468  2493  2494  2528  2531  2532  2582  2684  2686  2687  2690
MOV      411   413   414   415   418   419   421   422   423   424   425   435   446   451   459
         460   461   464   478   483   485   486   487   488   489   490   511   516   517   518
         520   521   523   539   544   546   549   550   574   578   582   584   585   586   587
         588   594   606   611   612   613   614   615   621   633   637   639   640   641   642
         643   644   648   649   650   651   662   669   674   675   676   677   678   679   683
         694   695   696   697   704   708   709   716   720   747   754   756   757   773   777
         779   780   781   782   787   788   789   800   805   807   809   811   812   824   825
         826   827   828   829   830   831   849   855   856   857   858   859   860   861   862
         863   865   866   869   871   872   873   875   876   877   881   883   905   906   907
         913   917   919   920   922   923   924   925   927   928   929   942   958   959   960
         977   982   983   988   989   990  1008  1010  1026  1027  1029  1033  1041  1046  1047
        1049  1054  1064  1069  1070  1072  1073  1087  1093  1097  1098  1100  1102  1103  1133
        1137  1139  1140  1142  1171  1176  1178  1180  1186  1212  1219  1223  1226  1227  1228
        1230  1231  1241  1242  1243  1244  1246  1251  1252  1265  1281  1286  1288  1298  1304
        1309  1310  1311  1313  1314  1315  1317  1319  1320  1325  1326  1327  1346  1348  1350
        1359  1365  1367  1368  1369  1370  1372  1373  1374  1377  1414  1419  1420  1421  1423
        1424  1427  1468  1470  1472  1473  1476  1510  1525  1527  1530  1531  1532  1534  1580
        1581  1589  1595  1596  1597  1600  1608  1610  1612  1627  1630  1632  1650  1657  1663
        1664  1665  1668  1676  1678  1680  1695  1698  1700  1718  1725  1731  1732  1733  1736
        1744  1746  1748  1763  1766  1768  1786  1790  1802  1806  1811  1814  1815  1817  1818
        1836  1837  1843  1849  1851  1853  1854  1867  1868  1872  1878  1883  1894  1900  1902
        1904  1905  1906  1907  1913  1920  1926  1933  1939  1940  1941  1947  1952  1953  1959
        1965  1970  1975  1976  1997  2002  2005  2007  2008  2010  2025  2030  2035  2037  2040
        2041  2046  2054  2060  2066  2078  2085  2091  2093  2095  2098  2102  2117  2133  2142
        2148  2150  2152  2155  2159  2173  2188  2195  2199  2202  2204  2206  2207  2208  2209
        2210  2240  2245  2248  2251  2260  2266  2267  2268  2269  2270  2272  2274  2276  2301
        2308  2319  2324  2334  2342  2353  2358  2369  2373  2376  2377  2378  2379  2382  2383
        2385  2391  2392  2410  2411  2434  2438  2441  2442  2443  2449  2450  2460  2464  2467
        2468  2469  2474  2475  2488  2492  2495  2497  2498  2516  2527  2533  2534  2536  2537
        2538  2549  2571  2573  2576  2579  2612  2620  2621  2623  2636  2637  2638  2643  2656
        2664  2675  2679  2680  2682  2683  2689  2699  2700  2702  2706  2718  2721  2728  2739
        2756  2768  2770  2773  2785  2786  2789  2800  2801
MOVB    1794  2313  2326  2347  2360  2574  2703  2708
NOP      700   725   731   737   966  1128  1146  1147  1161  1323  1330  1511  1512  2499  2583
        2584  2585
RESET    439   647   682   758   786   794   820  1125  1159  1798  1831  1862  1911  2408  2409
        2581  2790
ROL     2732  2733  2752  2753  2754
RTI     1820  2694
```

E06


```
RTS      2295  2616  2628  2707  2724  2746  2795  2797  2802
RTT      2009
SEC      2751
SOB       501   502   533   564   601   602   628   629   645   646   657   658   680   681   692
          693  1236  1261  1325  2401  2418  2477  2615  2624
SUB      2182  2184  2563  2564  2769
SWAB     2213  2217  2229  2233
TRAP      271
TST       416   426   440   443   455   500   532   554   563   600   627   656   659   691   694
          710   899   950  1015  1182  1204  1260  1271  1291  1568  1832  1856  1863  1909  1922
         1943  1955  2069  2098  2100  2106  2156  2157  2162  2212  2252  2294  2381  2400  2417
         2476  2614  2691  2719  2729  2735  2748
TSTB      996  2577  2709  2722  2740
.ABS        1
.ASCII   2590  2600  2602
.ASCIZ   2604  2605
.END     2806
.EVEN    2606
.LIST       1   307   408   434   450   478   510   538   577   605   632   668   703   715   746
          772   799   848   912   976  1092  1132  1170  1218  1280  1303  1358  1588  1656  1724
         1789  1805  1842  1871  1893  1932  1969  1996  2029  2059  2084  2141  2194  2244  2259
         2300  2333  2368  2433  2459  2487  2526
.MACR     408  1583
.NLIST      1   307   408   434   450   478   510   538   577   605   632   668   703   715   746
          772   799   848   912   976  1092  1132  1170  1218  1280  1303  1358  1588  1656  1724
         1789  1805  1842  1871  1893  1932  1969  1996  2029  2059  2084  2141  2194  2244  2259
         2300  2333  2368  2433  2459  2487  2526
.REM        1
.REPT     307  1379  1429  1478  1536
.TITLE      1
.WORD     332
```

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

```
#DFKTAA,DFKTAA.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DFKTAA.P11
RUN-TIME:  10 20 4 SECONDS
RUN-TIME RATIO:  77/36=2.1
CORE USED:  10K  (19 PAGES)
```

F06