

PDP11/34

MEMORY MANAGEMENT BASIC LOGIC
MD-11-DFKTA-A

EP DFKTA-A DL-A

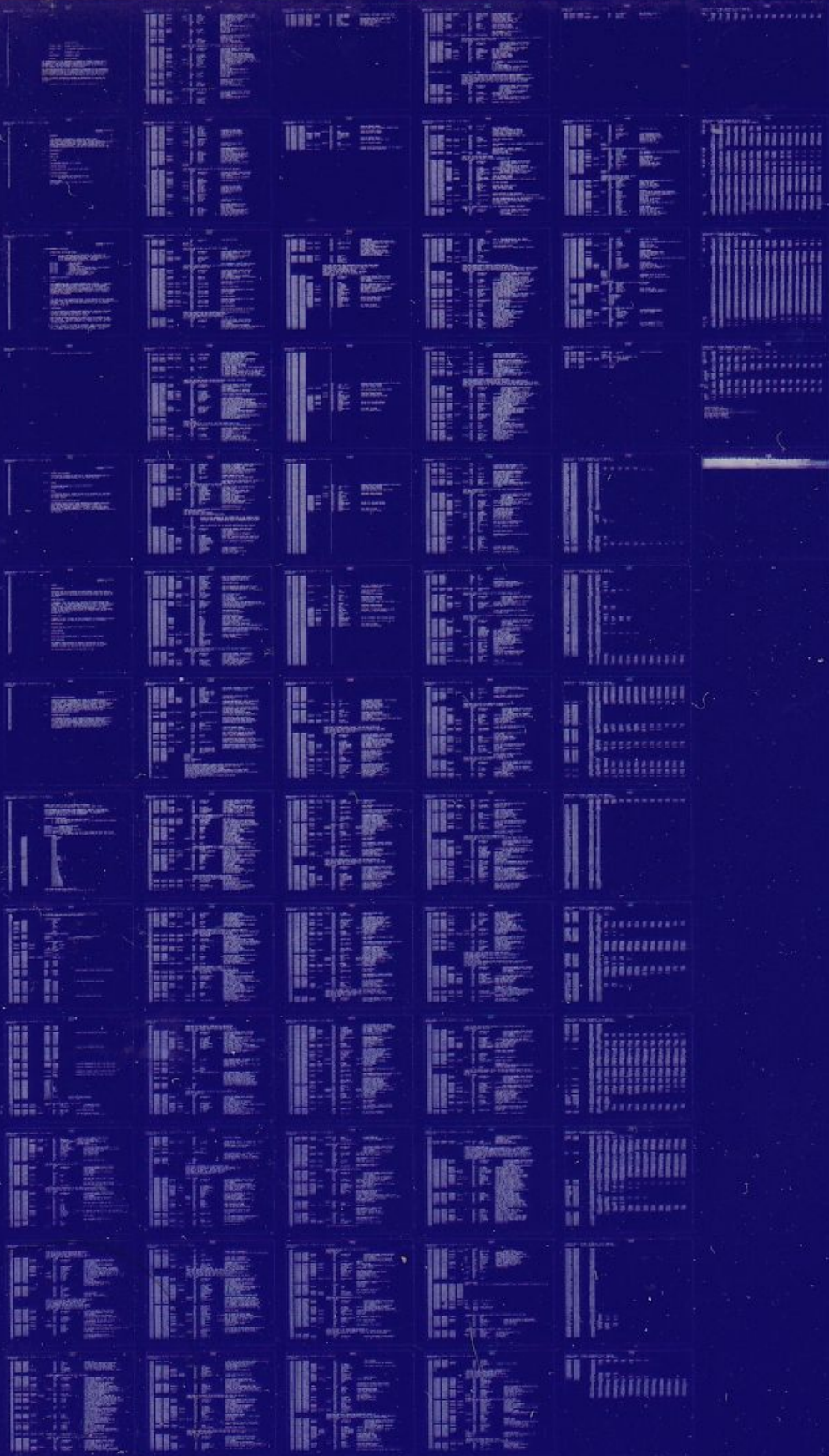
OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made in U.S.A.



11-11-76

.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

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH A SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

COPYRIGHT (C) 1975 BY DIGITAL EQUIPMENT CORPORATION

11-34 MEMORY MANAGEMENT BASIC LOGIC TEST

56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83

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 REGISTER 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 ENT 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 (00000). 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

100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139

E01

DFKTR.A MACY11 27(732) 09-SEP-76 17:12 PAGE 4
DFKTR.A.P11

140
141
142

COUNTER WHEN THE TRAP OR INTERRUPT OCCURRED.

...

143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174

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 SRD.

5.2.6 RVALL

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.

175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222

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.

223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300

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.

*

249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304

104400
000240
000000
000001
000002
000003
000004
000005
000006
000007
000006
000007
177776
177776
104006
000001
000002
000004
000010
000020
000040
000100
000200
000400
001000
002000
004000
010000
020000
040000
100000

;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, DOKTA. THIS VERSION
;HAS THE SOFTWARE SWITCH REGISTER CAPABILITIES AND HAS BEEN MODIFIED
;TO ACCOUNT 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

```

305 ;LOCATION, AND LOAD EACH LOCATION IMMEDIATELY FOLLOWING
306 ;A VECTOR ADDRESS WITH A HALT INSTRUCTION
307
308 ;LOAD VECTOR AREA
309      .=30
310 000030 016334      EMTSRV
311 000032 000340      340
312      .=34
313 000034 015544      SCOPEC
314 000036 000000      0
315      .=46
316 000046 015234      LOGIC
317 ;SOFTWARE SWITCH REGISTER
318      .=174
319 000174 000000      DISPREG: 0 ;SOFTWARE DISPLAY REGISTER
320 000176 000000      SWREG: 0 ;SOFTWARE SWITCH REGISTER
321
322 ;LOAD STARTING AREA
323      .=200
324 000200 000167 001000      JMP START
325      .=210
326 000210 000167 015242      JMP TESTX
327 ;LOAD DATA AREA
328      .=400
329 000400 000000      USTACK: 0
330      .=. +376
331 001000 000000      KSTACK: 0
332 001002 000000 000000 000000      .WORD 0,0,0,0
333 001010 000000
334 001012 123456      K123: 123456
335 001014 134567      K134: 134567
336 001016 177564      TCSR: 177564
337 001020 177566      TCSR: 177566
338 001022 000000      TEMP: 0
339
340 001024 177572      SR0: 177572 ;MEMORY MANAG. STATUS REGISTER ADDRESSES
341 001026 177573      SR0H: 177573
342 001030 177574      SR1: 177574
343 001032 177576      SR2: 177576
344
345
346 ;ADRTAB:
347 001034 177600      UPDR0: 177600 ;USER PAGE DESCRIPTOR REGISTERS
348 001036 177602      UPDR1: 177602
349 001040 177604      UPDR2: 177604
350 001042 177606      UPDR3: 177606
351 001044 177610      UPDR4: 177610
352 001046 177612      UPDR5: 177612
353 001050 177614      UPDR6: 177614
354 001052 177616      UPDR7: 177616
355
356 ;UPAR0: 177640 ;USER PAGE ADDRESS REGISTERS
357 001056 177642      UPAR1: 177642
358 001060 177644      UPAR2: 177644
359 001062 177646      UPAR3: 177646
360 001064 177650      UPAR4: 177650

```

K01

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 10
DFKTA.A.P11

361	001066	177652	UPARS:	177652	
362	001070	177654	UPAR6:	177654	
363	001072	177656	UPAR7:	177656	
364			.		
365	001074	172300	KPDR0:	172300	;KERNEL PAGE DESCRIPTOR REGISTERS
366	001076	172302	KPDR1:	172302	
367	001100	172304	KPDR2:	172304	
368	001102	172306	KPDR3:	172306	
369	001104	172310	KPDR4:	172310	
370	001106	172312	KPDR5:	172312	
371	001110	172314	KPDR6:	172314	
372	001112	172316	KPDR7:	172316	
373			.		
374	001114	172340	KPAR0:	172340	;KERNEL PAGE ADDRESS REGISTERS
375	001116	172342	KPAR1:	172342	
376	001120	172344	KPAR2:	172344	
377	001122	172346	KPAR3:	172346	
378	001124	172350	KPAR4:	172350	
379	001126	172352	KPAR5:	172352	
380	001130	172354	KPAR6:	172354	
381	001132	172356	KPAR7:	172356	
382		001132	ADREND=	.-2	
383			.		
384	001134	177600	PDRTAB:	177600	;STARTING ADDRESSES OF PDR'S FOR EACH MODE
385	001136	172300	PDREND:	172300	
386	001140	177640	PARTAB:	177640	;STARTING ADDRESSES OF PAR'S FOR EACH MODE
387	001142	172340		172340	
388					
389	001144	001074	STATAB:	KPDR0	;ADDRESS OF KERNEL TABLE OF PDR'S AND PAR'S
390	001146	000000		0	
391	001150	001034		UPDR0	;ADDRESS OF USER TABLE OF PDR'S AND PAR'S
392	001152	140000	STAREND:	140000	
393					
394					
395	001154	000000	STAPNT:	0	
396	001156	000000	PAGES:	0	
397	001160	000000	SAVEA:	0	
398	001162	000000	SAVEB:	0	
399	001164	000250	KTVEC:	250	
400	001166	000252	KTSTA:	252	
401	001170	100361	PDRM2:	100361	
402	001172	000000	FTITLE:	0	
403	001174	000000	TESTCT:	0	
404	001176	000000	BLOCKS:	0	
405	001200	177570	SP:	177570	;SWITCH REGISTER POINTER
406	001202	000000	DISPLAY:	0	;DISPLAY REGISTER POINTER
407					
408					
409			:SET UP	FOR START OF BASIC LOGIC TESTS	
410	001204	005037	START:	CLR 2#PS	;INITIALIZE STATUS
411	001210	012706		MOV #KSTACK,SP	;SETUP KERNEL STACK
412					
413	001214	013746		MOV 2#4,-(SP)	::SAVE ERROR VECTOR
414	001220	013746		MOV 2#6,-(SP)	
415	001224	012767		MOV 815,4	::SET UP TIME OUT VECTOR
416	001232	005777		TST 2SR	::TRY TO REFERENCE HARDWARE SWR

L01

```

417 001236 000407          BR      25          ;; BRANCH IF NO TIMEOUT TRAP OCCURS
418 001240 012767 000176 177732 18:  MOV     @SREG, SR      ;; POINT TO SOFTWARE SWP
419 001246 012767 000174 177726      MOV     @DISPREG, DISPLAY ;; POINT TO SOFTWARE DISPLAY REG
420 001254 022626          CMP     (SP)+, (SP)+    ;; RESTORE STACK
421 001256 012637 000006          MOV     (SP)+, @286    ;; RESTORE ERROR VECTOR
422 001262 012637 000004          MOV     (SP)+, @284
423 001266 012767 002000 014362  MOV     @2000, ICOUNT   ;; INITIALIZE ITERATION COUNT
424 001274 012767 001334 014360  MOV     @TEST1+2, RETURN ;; SETUP SCOPE AND ITERATION LOOP RETURN
425 001302 012767 000001 177664  MOV     @1, TESTCT     ;; INITIALIZE TEST COUNT
426 001310 005767 177656          TST     FTITLE         ;; DID TITLE PRINT
427 001314 001007          BNE    TEST1+2        ;; YES, START TEST
428 001316 004767 014460          JSR    PC, TYPE       ;; NO, PRINT TITLE
429 001322 015250          MTIT
430 001324 005267 177642          INC    FTITLE
431 001330 000401          BR     .+4           ;; SKP SCOPE INSTRUCTION
432
433
434          ;SRO AND SRI SHOULD BE INITIALIZED TO 0
435 TEST1:  SCOPE
436 001332 104400          MOV     @KSTACK, SP   ;; INITIALIZE KERNEL STACK POINTER
437 001334 012706 001000          JSR    PC, ORDER     ;; CHECK TEST SEQUENCE + INIT SRO
438 001340 004767 015040          I      ;; TEST NUMBER
439 001344 000001          HLT    1             ;; TEST EXECUTED OUT OF SEQUENCE
440 001346 104006          RESET          ;; ISSUE INIT
441 001348 000005          TST    @SRO         ;; CHECK SRO
442 001350 005777 177446          BEQ    .+4
443 001352 001401          HLT
444 001356 104006          TST    @SRI         ;; SRO WAS NOT INITIALIZED TO ZERO
445 001360 005777 177442          BEQ    .+4         ;; CHECK SRI
446 001362 001401          HLT
447 001366 104006          TST    @SRI         ;; SRI WAS NOT INITIALIZED TO ZERO
448 001370 001401          HLT
449 001372 012767 000010 014256  MOV     @10, ICOUNT   ;; DROP ITERATION COUNT SINCE RESET IS USED
450
451          ;CHECK READ/WRITE PROPERTIES OF ALL BITS IN SRO EXCEPT 0 AND 8
452          ;BY ROTATING A ONE THRU THE BIT POSITIONS BEING CHECKED
453 TEST2:  SCOPE
454 001400 104400          MOV     @KSTACK, SP   ;; INITIALIZE KERNEL STACK POINTER
455 001402 012706 001000          JSR    PC, ORDER     ;; CHECK TEST SEQUENCE + INIT SRO
456 001406 004767 014772          I      ;; TEST NUMBER
457 001412 000002          HLT    2             ;; TEST EXECUTED OUT OF SEQUENCE
458 001414 104006          TST    @SRO         ;; CHECK SRO INITIALLY
459 001416 005777 177402          BEQ    .+6
460 001422 001402          HLT
461 001424 104006          BR     EXIT2        ;; SRO NOT ZERO AT START OF TEST
462 001426 000422          MOV     @1, R0       ;; R0 CONTAINS BIT INDICATING POSITION BEING TESTED
463 001430 012700 000001          MOV     R0, R1
464 001434 010001          MOV     R1, R2
465 001436 010102          BIC    @401, R1      ;; DON'T SET THE BIT IN SRO IF IT'S BIT 0 OR BIT 8
466 001440 042701 000401          BIC    @17777, R2   ;; CLEAR THE BIT IN R2 IF IT SHOULDN'T SET IN SRO
467 001444 042702 017777          MOV     R1, @SRO
468 001448 010177 177350          CMP    R2, @SRO
469 001450 010177 177344          BEQ    .+4         ;; CHECK SRO
470 001454 020277 177344          HLT
471 001460 001401          ;; SRO INCORRECT WHEN VALUE IN R1
472 001462 104006          ;; WAS LOADED INTO IT
473
474          ASL    R0
475          BCC    LOOP2
476          CLR    @SRO
477
478          EXIT2:

```

MO1

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 12
DFKTA.A.P11

```

473
474
475
476
477
478 001474 104400
479 001476 012706 001000
480 001502 004767 014676
481 001506 000003
482 001510 104006
483 001512 012767 002000 014136
484 001520 004767 013644
485 001524 012703 001140
486 001530 012700 000002
487 001534 012301
488 001536 012702 000010
489 001542 012704 000001
490 001546 010411
491 001550 020411
492 001552 001401
493 001554 104006
494
495
496 001556 006304
497 001560 020427 010000
498 001564 001370
499 001566 005011
500 001570 005721
501 001572 077215
502 001574 077021
503
504
505
506
507
508
509
510 001576 104400
511 001600 012706 001000
512 001604 004767 014574
513 001610 000004
514 001612 104006
515 001614 004767 013650
516 001620 012703 001134
517 001624 012301
518 001626 012702 000010
519
520 001632 012700 000001
521 001636 010005
522 001640 046705 177324
523 001644 010011
524 001646 021105
525 001650 001401
526 001652 104006
527
528

```

```

;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
TEST3: SCOPE
        MOV     #KSTACK, SP
        JSR     PC, ORDER
        3
        HLT
        MOV     #2000, ICOUNT
        JSR     %7, CLRALL
        MOV     #PARTAB, R3
        MOV     #2, R0
LOOP3:   MOV     (R3)+, R1
        MOV     #10, R2
LOOP3A: MOV     #1, R4
LOOP3B: MOV     R4, @R1
        CMP     R4, @R1
        BEQ    .+4
        HLT
;INITIALIZE KERNEL STACK POINTER
;CHECK TEST SEQUENCE + INIT SRO
;TEST NUMBER
;TEST EXECUTED OUT OF SEQUENCE
;RESTORE ICOUNT
;INITIALIZE MEMORY MANAG. REGISTERS
;R3 POINTS TO TABLE OF PAR ADDRESSES
;R0 IS COUNTER OF STATES LEFT TO TEST
;PUT ADDRESS OF 1ST PAR IN SET IN R1
;R2 IS COUNTER OF PAR'S LEFT TO TEST IN SET
;R4 IS BIT OF PAR BEING TESTED
;SET BIT IN PAR
;CHECK PAR
;BRANCH IF OK
;PAR WHOSE ADDRESS IS IN R1
;FAILED WHEN THE VALUE IN R4
;WAS LOADED INTO IT
        ASL     R4
        CMP     R4, #10000
        BNE    LOOP3B
        CLR     @R1
        TST    (R1)+
        SOB    R2, LOOP3A
        SOB    R0, LOOP3
;MOVE POINTER
;TEST ALL PAR'S IN SET
;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
TEST4: SCOPE
        MOV     #KSTACK, SP
        JSR     PC, ORDER
        4
        HLT
        JSR     %7, CLRALL
        MOV     #PARTAB, R3
LOOP4:   MOV     (R3)+, R1
        MOV     #10, R2
LOOP4A: MOV     #1, R0
LOOP4B: MOV     R0, R5
        BIC    PDRM2, R5
        MOV     R0, @R1
        CMP     @R1, R5
        BEQ    .+4
        HLT
;INITIALIZE KERNEL STACK POINTER
;CHECK TEST SEQUENCE + INIT SRO
;TEST NUMBER
;TEST EXECUTED OUT OF SEQUENCE
;INITIALIZE MEMORY MANAG. REGISTERS
;LOAD ADDRESS OF 1ST PDR IN STATE
;USE R2 AS A COUNTER OF PDR'S
;LEFT TO TEST
;SETUP R0 TO ROTATE A BIT THRU
;R5 CONTAINS EXPECTED RESULTING CONTENTS OF PDR
;LOAD PDR
;CHECK RESULTING CONTENTS OF PDR
;PDR WHOSE ADDRESS IS IN R1
;WAS INCORRECT AFTER VALUE IN R0
;WAS LOADED INTO IT

```

529	001654	006300		ASL	R0		: ROTATE BIT
530	001656	103367		BCC	LOOP4B		: BRANCH IF NOT DONE WITH THIS PDR
531	001660	005011		CLR	@R1		: IF DONE WITH THIS PDR, CLEAR IT
532	001662	005721		TST	(R1)+		: MOVE POINTER TO ADDRESS NEXT PDR
533	001664	077216		SOB	R2, LOOP4A		: TEST ALL PDR'S IN THIS GROUP
534	001666	020327	001136	CMP	R3, #PDREND		: TEST ALL 2 GROUPS OF PDRS-USER, KERNEL
535	001672	003754		BLE	LOOP4		
536							
537							
538	001674	104400					: NO DUAL ADDRESSING TEST FOR PAR'S AND PDR'S
539	001676	012706	001000	TEST5:	SCOPE		
540	001702	004767	014476	MOV	#KSTACK, SP		: INITIALIZE KERNEL STACK POINTER
541	001706	000005		JSR	PC, ORDER		: CHECK TEST SEQUENCE + INIT SRO
542	001710	104006		5			: TEST NUMBER
543	001712	004767	013452	HLT			: TEST EXECUTED OUT OF SEQUENCE
544	001716	012701	001034	JSR	%7, CLRALL		: CLEAR ALL PAR'S AND PDR'S
545				MOV	#ADRTAB, R1		: R1 POINTS TO ADDRESS OF LOCATION
546	001722	012702	001034	LOPSAA:	MOV	#ADRTAB, R2	: LOADED WITH 1 BIT SET IN EACH 4 BITS
547							: R2 USED AS A POINTER TO CYCLE THRU
548							: ALL OTHER ADDRESSES OF PAR/PDR PAIR'S TO
549	001726	012703	000040				: CHECK FOR DUAL ADDRESSING
550	001732	012771	010421 000000	MOV	#32, R3		: R3 USED AS A COUNTER
551				MOV	#10421, @R1		: LOAD A PAR OR PDR - SET ONE BIT
552	001740	020201		LOPSB:	CMP	R2, R1	: IN EACH CHIP (4 BITS PER CHIP) IF R/W
553	001742	001406		BEQ	CONT5		: SKIP CHECKING THIS ADDRESS TO SEE IF
554	001744	005772	000000	TST	@R2		: IT'S A DUAL, SINCE IT WAS THE ONE LOADED
555							: OTHERWISE, CHECK TO SEE IF THIS
556							: REGISTER RESPONDED TO THE ADDRESS
557	001750	001403		BEQ	CONT5		: OF THE ONE LOADED AS A DUAL
558	001752	104006		HLT			: BRANCH IF OK
559							: DUAL ADDRESSING - ADDRESS POINTED
560							: TO BY R2 RESPONDED TO THE ADDRESS
561							: POINTED TO BY R1 IN AT LEAST ONE
562	001754	005072	000000				: 4 BIT SECTION (1 CHIP)
563	001760	005722		CONT5:	CLR	@R2	: REINITIALIZE FAULTY LOCATION
564	001762	077312		TST	(R2)+		: MOVE POINTER R2
565				SOB	R3, LOPSB		: CHECK ALL PAR'S AND PDR'S
566							: TO SEE IF THEY RESPONDED TO THE
567	001764	022701	001132	CMP	#ADREND, R1		: ADDRESS POINTED TO BY R1
568							: HAVE ALL ADDRESSES BEEN CHECKED
569	001770	001402		BEQ	DONESA		: FOR DUALS?
570	001772	005031		CLR	@R1)+		: YES - GO TO NEXT TEST
571	001774	000752		BR	LOPSAA		: NO - MOVE POINTER R1
572							: CHECK TO SEE IF ANY OTHER ADDRESS
573							: ALSO RESPONDS TO THE ADDRESS POINTED
574	001776	012767	000100 013652	DONESA:	MOV	#100, ICOUNT	: TO BY R1
575							: DROP ITERATION COUNT
576							
577	002004	104400					: SHOW THAT BYTE ADDRESSING OF PAR'S WORKS FOR HIGH AND LOW BYTES
578	002006	012706	001000	TEST6:	SCOPE		
579	002012	004767	014366	MOV	#KSTACK, SP		: INITIALIZE KERNEL STACK POINTER
580	002016	000006		JSR	PC, ORDER		: CHECK TEST SEQUENCE + INIT SRO
581	002020	104006		6			: TEST NUMBER
582	002022	012767	002000 013626	HLT			: TEST EXECUTED OUT OF SEQUENCE
583	002030	004767	013334	MOV	#2000, ICOUNT		: RESTORE ITERATION COUNT
584	002034	012703	001140	JSR	%7, CLRALL		: INITIALIZE MEMORY MANAG. REGISTERS
				MOV	#PARTAB, R3		: R3 POINTS TO TABLE OF PAR ADDRESSES

585	002040	012700	000002		MOV	R2,R0	:R0 IS COUNTER OF STATES LEFT TO TEST
586	002044	012301		LOOP6:	MOV	(R3)+,R1	:PUT ADDRESS OF 1ST PAR IN SET IN R1
587	002046	012702	000010		MOV	R10,R2	:R2 IS COUNTER OF PAR'S LEFT TO TEST IN SET
588	002052	012711	177777	LOOP6A:	MOV	R-1,R1	:SET UP PAR BEING TESTED
589	002056	105011			CLRB	R1	:CLEAR LOW BYTE OF PAR
590	002060	022711	007400		CMP	R7400,R1	:CHECK PAR
591	002064	001401			BEG	.+4	:BRANCH IF OK
592	002066	104006			HLT		:DATOB TO PAR WHOSE ADDRESS IS IN
593							:R1 FAILED
594	002070	012711	177777		MOV	R-1,R1	:SET UP PAR TO TEST HIGH BYTE
595	002074	105061	000001		CLRB	1(R1)	:CLEAR HIGH BYTE
596	002100	022711	000377		CMP	R377,R1	:CHECK PAR
597	002104	001401			BEG	.+4	
598	002106	104006			HLT		:DATOB TO HIGH BYTE OF PAR WHOSE
599							:ADDRESS IS IN R1 FAILED
600	002110	005721			TST	(R1)+	:MOVE POINTER
601	002112	077221			SQB	R2,LOOP6A	:TEST ALL PAR'S IN SET
602	002114	077025			SQB	R0,LOOP6	:TEST ALL 2 REGISTER SETS
603							
604							
605	002116	104400					:SHOW THAT BYTE ADDRESSING OF PDR'S WORKS FOR HIGH AND LOW BYTES
606	002120	012706	001000	TEST7:	SCOPE		
607	002124	004767	014254		MOV	RKSTACK,SP	:INITIALIZE KERNEL STACK POINTER
608	002130	000007			JSR	PC,ORDER	:CHECK TEST SEQUENCE + INIT SRO
609	002132	104006					:TEST NUMBER
610	002134	004767	013230		HLT		:TEST EXECUTED OUT OF SEQUENCE
611	002140	012703	001134		JSR	X7,CLRALL	:INITIALIZE MEMORY MANAG. REGISTERS
612	002144	012700	000002		MOV	RPORTAB,R3	:R3 POINTS TO TABLE OF PDR ADDRESSES
613	002150	012301		LOOP7:	MOV	R2,R0	:R0 IS COUNTER OF STATES LEFT TO TEST
614	002152	012702	000310		MOV	(R3)+,R1	:PUT ADDRESS OF 1ST PDR IN SET INTO R1
615	002156	012711	177777	LOOP7A:	MOV	R10,R2	:R2 IS COUNTER OF PDR'S LEFT TO TEST IN SET
616	002162	105011			MOV	R-1,R1	:SET UP PDR BEING TESTED
617	002164	022711	077400		CLRB	R1	:CLEAR LOW BYTE OF PDR
618	002170	001401			CMP	R77400,R1	:CHECK PDR
619	002172	104006			BEG	.+4	:BRANCH IF OK
620					HLT		:DATOB TO PDR WHOSE ADDRESS IS
621	002174	012711	177777				:IN R1 FAILED
622	002200	105061	000001		MOV	R-1,R1	:SET UP PDR TO TEST HIGH BYTE
623	002204	022711	000016		CLRB	1(R1)	:CLEAR HIGH BYTE
624	002210	001401			CMP	R16,R1	:CHECK PDR
625	002212	104006			BEG	.+4	
626					HLT		:DATOB TO HIGH BYTE OF PDR WHOSE
627	002214	005721					:ADDRESS IS IN R1 FAILED
628	002216	077221			TST	(R1)+	:MOVE POINTER
629	002220	077025			SQB	R2,LOOP7A	:TEST ALL PDR'S IN SET
630					SQB	R0,LOOP7	:TEST ALL 2 REGISTER SETS
631							
632	002222	104400					:INIT SHOULD HAVE NO EFFECT ON PAR'S
633	002224	012706	001000	TEST10:	SCOPE		
634	002230	004767	014150		MOV	RKSTACK,SP	:INITIALIZE KERNEL STACK POINTER
635	002234	000010			JSR	PC,ORDER	:CHECK TEST SEQUENCE + INIT SRO
636	002236	104006					:TEST NUMBER
637	002240	012767	000010		HLT		:TEST EXECUTED OUT OF SEQUENCE
638	002246	005067	000104		MOV	R10,ICOUNT	:DROP ITERATION COUNT
639	002252	012704	005252		CLR	TST10F	
640	002256	012703	001140	TST10:	MOV	R5252,R4	
					MOV	RPARTAB,R3	

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 15
DFKTA.A.P11

641	002262	012700	000002		MOV	#2,R0	
642	002266	012301		LOOP10:	MOV	(R3)+,R1	
643	002270	012702	000010		MOV	#10,R2	:COUNTER TO LOAD PAR'S
644	002274	010421		LOP10A:	MOV	R4,(R1)+	:LOAD PAR WITH PATTERN
645	002276	077202			SOB	R2,LOP10A	:LOAD ALL 16 IN THIS SET
646	002300	077006			SOB	R0,LOOP10	:INITIALIZE ALL 2 SETS
647	002302	000005			RESET		:ISSUE INIT
648	002304	012703	001140		MOV	#PARTAB,R3	
649	002310	012700	000002		MOV	#2,R0	
650	002314	012301		LOP10B:	MOV	(R3)+,R1	
651	002316	012702	000010		MOV	#10,R2	:COUNTER TO CHECK PAR'S
652	002322	020411		LOP10C:	CMP	R4,R1	:CHECK DATA
653	002324	001401			BEQ	.+4	
654	002326	104006			HLT		:PAR WHOSE ADDRESS IS IN R1
655							:WAS INCORRECT AFTER INIT
656	002330	005721			TST	(R1)+	:MOVE POINTER
657	002332	077205			SOB	R2,LOP10C	:TEST ALL 8 PAR'S IN THIS SET
658	002334	077011			SOB	R0,LOP10B	:TEST ALL 2 REGISTER SETS
659	002336	005767	000014		TST	TST10F	:CHECK FOR BOTH PATTERNS USED
660	002342	001006			BNE	EXIT10	:IF DONE, GO TO NEXT TEST
661	002344	005267	000006		INC	TST10F	:IF NOT, SET FLAG
662	002350	012704	002525		MOV	#2525,R4	:LOAD OTHER PATTERN
663	002354	000740			BR	TST10	:REPEAT TEST WITH 2ND PATTERN
664	002356	000000		TST10F:	0		
665	002360			EXIT10:			
666							
667							
668	002360	104400					:INIT SHOULDN'T CLEAR OR SET ANY OF THE R/W BITS IN THE PDR'S
669	002362	012706	001000	TEST11:	SCOPE		
670	002366	004767	014012		MOV	#KSTACK,SP	:INITIALIZE KERNEL STACK POINTER
671	002372	000011			JSR	PC,ORDER	:CHECK TEST SEQUENCE + INIT SRO
672	002374	104006			LI		:TEST NUMBER
673	002376	005067	000104		HLT		:TEST EXECUTED OUT OF SEQUENCE
674	002402	012704	025012		CLR	TST11F	
675	002406	012703	001134	TST11:	MOV	#25012,R4	:LOAD PATTERN IN R4
676	002412	012700	000002		MOV	#PDRTAB,R3	
677	002416	012301			MOV	#2,R0	
678	002420	012702	000010	LOOP11:	MOV	(R3)+,R1	:COUNTER TO LOAD PDR'S
679	002424	010421			MOV	#10,R2	:LOAD PDR WITH PATTERN
680	002426	077202		LOP11A:	MOV	R4,(R1)+	:LOAD ALL 8 IN THIS SET
681	002430	077006			SOB	R2,LOP11A	:INITIALIZE ALL 2 SETS
682	002432	000005			SOB	R0,LOOP11	:ISSUE INIT
683	002434	012703	001134		RESET		
684	002440	012700	000002		MOV	#PDRTAB,R3	
685	002444	012301			MOV	#2,R0	
686	002446	012702	000010	LOP11B:	MOV	(R3)+,R1	
687	002452	020411			MOV	#10,R2	:COUNTER TO CHECK PDR'S
688	002454	001401		LOP11C:	CMP	R4,R1	:CHECK DATA
689	002456	104006			BEQ	.+4	
690					HLT		:PDR WHOSE ADDRESS IS IN R1
691	002460	005721					:WAS INCORRECT AFTER INIT
692	002462	077205			TST	(R1)+	:MOVE POINTER
693	002464	077011			SOB	R2,LOP11C	:TEST ALL 8 PDR'S IN THIS SET
694	002466	005767	000014		SOB	R0,LOP11B	:TEST ALL 2 REGISTER SETS
695	002472	001006			TST	TST11F	:CHECK FOR BOTH PATTERNS USED
696	002474	005267	000006		BNE	EXIT11	:IF DONE, GO TO NEXT TEST
					INC	TST11F	:IF NOT, SET FLAG

753									: AND PAGE LENGTH ERRORS IF ACCESSED
754	002732	012777	001006	176134	MOV	#1006, 2KPDR0			: 3 BLOCKS OF KERNEL PDR0 MUST BE MAPPED
755									: TO ALLOW TRAPS AND ABORTS
756	002740	012767	000010	012710	MOV	#10, ICOUNT			: DROP THE ITERATION COUNT
757	002746	012777	000400	176050	MOV	#400, 2SRO			: TURN ON DESTINATION ONLY RELOCATION
758	002754	000005			RESET				: SHOULD CLEAR DEST ONLY BIT, AND A
759									: SOLID PLACE TO START
760									: IF THE FETCH IS RELOCATED
761									: THIS WILL GIVE A PL ABORT
762	002756	032777	000400	176040	BIT	#400, 2SRO			: IF MEMORY MANAG. STILL ON, THIS SHOULD CAUSE
763	002764	001401			BEQ	+.4			: PL AND NR ERRORS
764	002766	000000			HALT				: IF MEMORY MANAG. IS OFF, BIT 8 OF SRO READS
765									: AS STILL SET OR ANOTHER BIT IS INCORRECT
766									: IF MEMORY MANAG. IS ON, NO NR OR SL ABORT
767									: OCCURRED AND RESET FAILED TO TURN MEMORY MANAG.
768	002770	005077	176030		CLR	2SRO			
769									
770									: SHOW THAT DESTINATION ONLY RELOCATION DOESN'T RELOCATE THE SOURCE
771									: ADDRESS AND DOES RELOCATE THE DESTINATION
772	002774	104400			TEST15: SCOPE				
773	002776	012706	001000		MOV	#KSTACK, SP			: INITIALIZE KERNEL STACK POINTER
774	003002	004767	013376		JSR	PC, ORDER			: CHECK TEST SEQUENCE + INIT SRO
775	003006	000015			15				: TEST NUMBER
776	003010	104006			HLT				: TEST EXECUTED OUT OF SEQUENCE
777	003012	012767	000010	012636	MOV	#10, ICOUNT			: KEEP THE NUMBER OF LOOPS DOWN
778	003020	004767	012344		JSR	%7, CLRALL			
779	003024	012777	000001	176062	MOV	#1, 2KPAR0			: OFFSET KERNEL PAR/PDR PAIR 0 ONE BLOCK FROM BANK
780	003032	012777	077406	176034	MOV	#77406, 2KPDR0			
781	003040	012701	003112		MOV	#DATA16, R1			: LOAD A BANK 0 ADDRESS
782	003044	012777	000400	175752	MOV	#400, 2SRO			: TURN ON DESTINATION ONLY RELOCATION
783	003052	021111			CMP	2R1, 2R1			: THIS TEST WILL FAIL IF BOTH ARE
784	003054	001001			BNE	+.4			: RELOCATED OR BOTH ARE NOT RELOCATED
785	003056	000000			HALT				: SOURCE AND DESTINATION BOTH ADDRESSED SAME LOCA
786	003060	000005			RESET				: TURN OFF DESTINATION-ONLY RELOCATION
787	003062	012701	003012		MOV	#DATA16-100, R1			: LOAD DESTINATION ADDRESS MINUS RELOCATION FACTO
788	003066	012702	003112		MOV	#DATA16, R2			: LOAD SOURCE ADDRESS
789	003072	012777	000400	175724	MOV	#400, 2SRO			: TURN ON DESTINATION-ONLY RELOCATION
790	003100	021211			CMP	2R2, 2R1			: USE SAME INSTRUCTION AND ADDRESS
791	003102	001401			BEQ	+.4			: MODES AS BEFORE
792	003104	000000			HALT				: DESTINATION NOT RELOCATED OR INCORRECTLY
793									: RELOCATED OR SOURCE RELOCATED
794	003106	000005			RESET				: TURN OFF RELOCATION
795	003110	000401			BR	+.4			
796	003112	132465			DATA16: 132465				
797									: SHOW THAT A DATO OF 0 TO BIT 8, SRO THRU KERNEL PAGE 7 WILL
798									: CLEAR THE DESTINATION ONLY RELOCATION BIT AND TURN OFF DESTINATION ONLY RELOCATION
799	003114	104400			TEST16: SCOPE				
800	003116	012706	001000		MOV	#KSTACK, SP			: INITIALIZE KERNEL STACK POINTER
801	003122	004767	013256		JSR	PC, ORDER			: CHECK TEST SEQUENCE + INIT SRO
802	003126	000016			16				: TEST NUMBER
803	003130	104006			HLT				: TEST EXECUTED OUT OF SEQUENCE
804	003132	004767	012232		JSR	%7, CLRALL			: INITIALIZE
805	003136	012777	000001	175750	MOV	#1, 2KPAR0			: MAP KERNEL PAR/PDR PAIR 0
806									: TO BANK 0 OFFSET BY 1 PAGE
807	003144	012777	077406	175722	MOV	#77406, 2KPDR0			: USED TO PROVE MEMORY MANAG. IS
808									: TURNED OFF AFTER CLEARING BIT 8, SRO

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	SRO, R2		: SETUP R2 TO A OF 55 SRO
812	003166	012777	000400	175630	MOV	#400, ASRO		: TURN ON DESTINATION ONLY RELOCATION
813	003174	005012			CLR	ASR2		: CLEAR SRO THRU KERNEL PAR/PDR PAIR7
814	003176	021111			CMP	ASR1, ASR1		: 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, ASRO		: SHOW THAT BIT 8, SRO 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 DATA OF 0 TO BIT 8, SRO THRU USER PAGE 7
822								: WILL TURN OFF DESTINATION - ONLY PAGING
823	003220	004767	012144		JSR	X7, CLRALL		: INITIALLY CLEAR ALL PAR/PDR PAIRS
824	003224	012777	000001	175622	MOV	#1, ASR0		: MAP USER 0 TO
825	003232	012777	077406	175574	MOV	#77406, ASUPDR0		: 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, ASUPAR7		: MAP USER 7 TO THE
828	003252	012777	077406	175572	MOV	#77406, ASUPDR7		: EXTERNAL BANK
829	003260	016702	175540		MOV	SRO, R2		: SETUP R2 TO ADDRESS SRO
830	003264	012737	140000	177776	MOV	#140000, AS#PS		: SET MODE TO USER
831	003272	012777	000400	175524	MOV	#400, ASRO		: TURN ON DESTINATION - ONLY PAGING
832	003300	005012			CLR	ASR2		: CLEAR SRO THRU USER ASR7
833	003302	021111			CMP	ASR1, ASR1		: 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								
843								
844								
845								
846								
847								
848	003310	104400						
849	003312	012706	001000		TEST17:	SCOPE		
850	003316	004767	013062		MOV	#KSTACK, SP		: INITIALIZE KERNEL STACK POINTER
851	003322	000017			JSR	PC, ORDER		: CHECK TEST SEQUENCE + INIT SRO
852	003324	104006			17			: TEST NUMBER
853	003326	004767	012036		HLT			: TEST EXECUTED OUT OF SEQUENCE
854	003332	004767	012056		JSR	X7, CLRALL		: INITIALIZE
855	003336	013767	017700	175614	JSR	X7, RWALL		: MAKE ALL PAR/PDR PAIR'S RW, BANK 0, 4K
856	003344	013767	017776	175610	MOV	AS#17700, SAVEA		: SAVE CONTENTS OF LOCATIONS TO BE USED
857	003352	012737	123456	017700	MOV	AS#17776, SAVEB		
858	003360	012737	134567	017776	MOV	#123456, AS#17700		: SET UP LOCATIONS TO BE REFERENCED
859	003366	012703	001012		MOV	#134567, AS#17776		
860	003372	012704	001014		MOV	#K123, R3		
861	003376	012767	000100	012252	MOV	#K134, R4		
862	003404	012737	140000	177776	MOV	#100, ICOUNT		: CHANGE ITERATION COUNT
863	003412	012706	000400		MOV	#140000, AS#PS		: CHANGE TO USER
864	003416	005037	177776		MOV	#USTACK, SP		: SET UP USER STACK POINTER
					CLR	AS#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	R20, R0	; ADDRESS TABLE FOR NEW STATE
868	003440	062767	000002	175506		ADD	R2, STAPNT	
869	003446	017737	175502	177776		MOV	@STAPNT, @#PS	; SET UP NEW STATE
870	003454	062767	000002	175472		ADD	R2, STAPNT	
871	003462	012767	000010	175466		MOV	R8, PAGES	; SET UP COUNTER OF ASR'S LEFT TO TEST
872	003470	012770	007600	000016		MOV	R76, 0, @16(R0)	; SET UP SEGMENTED REFERENCE TO SRO
873	003476	016705	175322			MOV	SRO, R5	; USED TO TURN DESTINATION - ONLY PAGING OFF
874	003502	005001				CLR	R1	
875	003504	012702	000076			MOV	R76, R2	
876	003510	012767	000200	175460	PAG20:	MOV	R128, BLOCKS	; SET UP BLOCK COUNT
877	003516	012770	000177	000000		MOV	R177, @1(R0)	; SET UP PAR
878	003524	022767	000001	175424		CMF	R1, PAGES	; IS THIS PAGE ?? (WAS USED
879								; FOR REFERENCE TO SRO)
880	003532	001005				BNE	BLK20	; IF NOT, BRANCH
881	003534	012770	007600	177776		MOV	R76, 0, @-2(R0)	; YES, SET UP PAGE 6 FOR REFERENCES TO SRO
882	003542	042705	020000			BIC	R20000, R5	; CHANGE R5 TO POINT TO SRO THRU PAR/PDR PAIRS
883	003546	012777	000400	175250	BLK20:	MOV	R400, @SRO	; TURN ON DESTINATION ONLY PAGING
884	003554	021311				CMF	R3, R1	; CK BOTTOM PAGE BOUNDARY
885	003556	001401				BEQ	.+4	
886	003560	000000				HALT		; RELOCATION FAILED
887	003562	021412				CMF	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	R100, R1	; SET UP R1 AND R2 TO REFERENCE
893	003602	062702	000100			ADD	R100, 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:	CMF	STAPNT, #STAEND	; CHECK FOR ALL STATES TESTED
902	003640	003673				BLE	STAT20	; IF NOT, BRANCH
903	003642	005037	177776			CLR	@#PS	; IF DONE, REINITIALIZE
904	003646	005077	175152			CLR	@SRO	
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								
911								
912	003700	104400						; SHOW THAT THE INSTRUCTIONS USED IN THE NEXT TEST RELOCATE CORRECTLY IN
913	003702	012706	001000					; DESTINATION ONLY RELOCATION
914	003706	004767	012472		TEST20:	SCOPE		
915	003712	000020				MOV	#KSTACK, SP	; INITIALIZE KERNEL STACK POINTER
916	003714	104006				JSR	PC, ORDER	; CHECK TEST SEQUENCE + INIT SRO
917	003716	012767	002000	011732		20		; TEST NUMBER
918	003724	004767	011440			HLT		; TEST EXECUTED OUT OF SEQUENCE
919	003730	012777	000001	175156		MOV	R2000, ICOUNT	; RESTORE ITERATION COUNT
920	003736	012777	077406	175130		JSR	%7, CLALL	; CLEAR ALL MEMORY MANAG. REGISTERS
						MOV	R1, @KPAR0	; OFFSET KERNEL I-SPACE PAGE 0
						MOV	R77406, @KPOR0	; 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	SRO,R1		:SETUP R1 TO REFERENCE SRO
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,ASRO		: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	ARI		: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						
963	004136	000000						
964	004140	000000						
965	004142	004036						
966	004144	000240						
967								
968								
969								
970								
971								
972								
973								
974								
975								
976	004146	104400						

DST21A: 0
DST21B: 0
DST21C: 0
AD21C: DST21B-100
EXIT21: NOP

:TEST OF RELOCATION ADDERS - CHECK CORRECT PROPAGATION OF CARRY, AND CORRECT
:OUTPUT FOR EACH POSSIBLE COMBINATION FOR EACH BIT POSITION
:USE DESTINATION - ONLY RELOCATION, KERNEL
:TEST BY USING THE NECESSARY VALUE IN KERNEL PAR 1, WITH THE SECOND
:NECESSARY VALUE BEING THE VIRTUAL ADDRESS REFERENCE TO KERNEL PAR 1
:CHECK THE RESULTING PHYSICAL ADDRESS BY READING THE CONTENTS OF THE LOCATION,
:AND WRITING INTO THE LOCATION
:NOTE THAT THIS INCLUDES CHECKS OF ADDRESS WRAP AROUND
TEST21: SCOPE

977	004150	012706	001000		MOV	#KSTACK, SP	: INITIALIZE KERNEL STACK POINTER
978	004154	004767	012224		JSR	PC, ORDER	: CHECK TEST SEQUENCE + INIT SRO
979	004160	000021			21		: TEST NUMBER
980	004162	104006			HLT		: TEST EXECUTED OUT OF SEQUENCE
981	004164	004767	011200		JSR	X7, CLRALL	: CLEAR ALL MEMORY MANAG. REGISTERS
982	004170	012777	077406	174676	MOV	#77406, #KPDOR	: MAP KERNEL 0 TO BANK 0, 4K, RW
983	004176	012777	077406	174672	MOV	#77406, #KPDOR1	: 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, #SRO	: 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 77777 THRU KERNEL 1
995	004246	005077	174552		CLR	#SRO	: 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	#SRO	: 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, #SRO	: 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	#SRO	: 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	#SRO	: 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	CNT22C: 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, #SRO	: 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, @020100	: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	CNT220	: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 ADDR22C
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, @037700	: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, @037700	: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 ADDR22D
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	@017676, -(SP)	:SAVE CONTENTS OF LOCATION TO BE
1071								:REFERENCED
1072	004560	012737	076767	017676		MOV	@76767, @017676	: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, @037777	: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	@037777	:WRITE SAME LOCATION
1078	004610	005077	174210			CLR	@SR0	:TURN OFF MEMORY MANAG.
1079	004614	022737	000367	017676		CMP	@367, @017676	: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 ADDR22E
1087	004636	012637	017676		END22E:	MOV	(SP)+, @017676	:RESTORE LOCATION REFERENCED
1088								

```

1089                                     ;SHOW THAT SETTING SRO(0) TURNS ON FULL RELOCATION
1090                                     ;SHOW THAT ALL ADDRESS CALCULATIONS ARE RELOCATED
1091                                     ;SHOW THAT INIT CLEARS SRO(0) AND TURNS OFF RELOCATION
1092 004642 104400 TEST22: SCOPE
1093 004644 012706 001000      MOV      #KSTACK,SP
1094 004650 004767 011530      JSR      PC,ORDER
1095 004654 000022          22
1096 004656 104006          HLT
1097 004660 012767 000010 010770      MOV      #10,ICOUNT
1098 004666 004767 010476      JSR      %7,CLALL
1099 004672 012777 000001 174214      MOV      #1,%KPAR0
1100 004700 012777 077406 174166      MOV      #77406,%KPD0
1101 004706 004767 011556      JSR      PC,KERN7
1102 004712 012767 052525 012772      MOV      #52525,DESTAD
1103 004720 012777 000001 174076      MOV      #1,%SRO
1104 004726 000000      ADD23: HALT
1105 004730 000000          HALT
1106 004732 000000          HALT
1107 004734 000000          HALT
1108 004736 000000          HALT
1109 004740 000000          HALT
1110 004742 032777 000001 174054      BIT      #1,%SRO
1111                                     ;WHEN MEMORY MANAG. IS TURNED OFF, NEXT
1112 004750 001401      BEQ      .+4
1113 004752 104006          HLT
1114                                     ;FETCH SHOULD BE FROM HERE -
1115 004754 005077 174044      CLR      %SRO
1116 004760 000432      BR      EXIT23
1117 005026 022737 052525 017612      CMP      #52525,%DESTAD-100
1118                                     ;WHEN MEMORY MANAG. IS TURNED ON, NEXT
1119                                     ;INSTRUCTION EXECUTED SHOULD
1120                                     ;BE HERE - OK RELOCATION OF SOURCE
1121 005034 001401      BEQ      .+4
1122 005036 000000          HALT
1123                                     ;AND DESTINATION CALCULATIONS
1124                                     ;RELOCATION FAILED IN A SOURCE OR
1125                                     ;DESTINATION ADDRESS CALCULATION
1126 005040 000005      RESET
1127 005042 000000          HALT
1128 005044 000777          BR      .
1129 005046 000240      EXIT23: NOP
1130                                     ;SHOW THAT A DATA OF 0 TO SRO(0) WILL CLEAR SRO(0) AND
1131                                     ;TURN OFF RELOCATION
1132 005050 104400 TEST23: SCOPE
1133 005052 012706 001000      MOV      #KSTACK,SP
1134 005056 004767 011322      JSR      PC,ORDER
1135 005062 000023          23
1136 005064 104006          HLT
1137 005066 012767 002000 010562      MOV      #2000,ICOUNT
1138 005074 004767 012270      JSR      %7,CLALL
1139 005100 012777 000001 174006      MOV      #1,%KPAR0
1140 005106 012777 077406 173760      MOV      #77406,%KPD0
1141 005114 004767 011350      JSR      PC,KERN7
1142 005120 012777 000001 173676      MOV      #1,%SRO
1143 005126 000000      ADD24: HALT
1144 005130 000000          HALT
1145                                     ;WHEN MEMORY MANAG. IS TURNED ON, SHOULD
1146                                     ;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	01,2SR0		;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 01,2SR0
1152	005152	000433		BR	EXIT24		
1153							
1154							
1155	005226	005226	000001 173570	BIC	=ADD24+100 01,2SR0		;WHEN MEMORY MANAG. IS TURNED ON, SHOULD
1156		042777					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							
1164							
1165							
1166							
1167							
1168							
1169							
1170	005244	104400		TEST24: SCOPE			
1171	005246	012706	001000	MOV	0KSTACK, SP		;INITIALIZE KERNEL STACK POINTER
1172	005250	004767	011126	JSR	PC, ORDER		;CHECK TEST SEQUENCE + INIT SR0
1173	005252	000024		MOV			TEST NUMBER
1174	005260	104006		HLT			TEST EXECUTED OUT OF SEQUENCE
1175	005262	004767	010102	JSR	X7, CLRALL		;CLEAR ALL MEMORY MANAG. REGISTERS
1176	005266	012777	077406 173600	MOV	077406, 2KPOR0		MAP KERNEL 0 TO BANK 0, RM 4K
1177	005274	004767	011170	JSR	PC, KER47		MAP KERNEL PAR/PDR 7 TO EXT BANK
1178	005300	012777	005334 173656	MOV	0INT25, 2KTVEC		SETUP RETURN VECTOR
1179	005306	005077	173654	CLR	2KSTRA		
1180	005312	012704	020000	MOV	020000, RM		USE RM TO REFERENCE NR KERNEL 1
1181	005316	005277	173502	INC	2SR0		TURN ON MEMORY MANAG.
1182	005322	005724		ADR25: TST	(RM)+		REFERENCE NR KERNEL 1
1183	005324	000000		ADR25A: HALT			SHOULD HAVE RESTARTED ALREADY
1184	005326	005077	173472	CLR	2SR0		TURN OFF MEMORY MANAG.
1185	005332	000442		BR	DON25		
1186	005334	017701	173464	INT25: MOV	2SRC, R1		SAVE CONTENTS OF SR0
1187	005340	005377	173460	DEC	2SR0		TURN OFF MEMORY MANAG.
1188	005344	022701	100003	MOV	0100003, 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	ADR25: CMP	0ADR25, 2SR2		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	2SR2		TRY TO WRITE INTO SR2
1197	005372	022777	005322 173432	ADR25: CMP	0ADR25, 2SR2		SR2 SHOULD BE READ ONLY
1198	005400	001401		BEQ	.+4		
1199	005402	104006		HLT			SR2 NOT READ ONLY
1200	005404	022777	077506 173462	ADR25: CMP	077506, 2KPOR0		

1201	005412	001401			BEQ	.+4		
1202	005414	104006			HLT			; KERNEL PDR 0 INCORRECT ; W BIT SHOULD HAVE BEEN SET BY THE STACK WRITE
1203								
1204	005416	005777	173454		TST	@KPDRI		
1205	005422	001401			BEQ	.+4		
1206	005424	104006			HLT			; KERNEL PDR 1 INCORRECT
1207	005426	021627	005324		CHP	(R6), @ADR25A		; CHECK VALUE PUSHED ON STACK
1208	005432	001401			BEQ	.+4		
1209	005434	104006			HLT			; INCORRECT VALUE ON STACK
1210	005436	022626			CHP	(R6)+, (R6)+		; RESTORE STACK
1211	005440	005077	173522		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 SRO
1221	005464	000025			ZS			; 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	X7, @MALL		; 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, @SPS		; SET MODE TO USER
1228	005522	012706	000400		MOV	@USTACK, R6		; SET UP USER STACK
1229	005526	005037	177776		CLR	@SPS		; REINITIALIZE STATUS TO KERNEL MODE
1230	005532	012704	001034		MOV	@ADR25A, R4		
1231	005536	012705	000010		MOV	@10, R5		; LOAD R4 WITH ADDRESS OF ADR TABLE
1232	005542	022734	077406		LOP31A: CHP	@77406, @R4)+		; INIT COUNTER OF PDR'S LEFT TO CHECK
1233	005548	001401			LOP31B: 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			S08	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		CHP	R4, @ADREND		
1239	005564	003000			BGT	CNT31A		; BRANCH IF DONE
1240	005566	000000			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)+, @SPS		; SET UP STATUS FOR CURRENT MODE
1245	005606	005277	173212		LOP31D: INC	@SRO		; TURN ON MEMORY MANAG.
1246	005612	011212			MOV	(R2), (R2)		; REFERENCE PAGE TO SET W BIT
1247	005614	005077	173204		CLR	@SRO		; 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	@ADR25A, 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		

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 26
DFKTA.A.P11

```
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, 2(R1) ; CLEAR W BIT VIA DATO TO PDR
1266 005704 032771 000100 000000 BIT #100, 2(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 025000 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 #SRO ; REINITIALIZE SRO
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 SRO
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, 2KPOR1 ; 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, 2KTVEC ; SETUP ABORT RETURN
1289 006002 005077 173160 CLR 2KTSTA
1290 006006 005277 173012 INC #SRO ; TURN ON MEMORY MANAG.
1291 006012 005737 030000 TST 2#30000 ; REFERENCE NR KERNEL 1 - SHOULD ABORT
1292 006016 000000 HALT ; NO NR ABORT
1293 006020 022777 140003 172776 RET33: CMP #140003, 2SRO ; CHECK SRO
1294 006026 001401 BEQ .+4
1295 006030 104006 HLT ; SRO INCORRECT - SHOULD SHOW KERNEL
1296 ; PAGE 1, AND BOTH NR + PL ERRORS SET
1297 006032 005077 172766 CLR #SRO
1298 006036 016777 173124 173120 MOV #KTSTA, 2KTVEC ; 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 SRO
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, 2#PS
1311 006100 012706 000100 MOV #100, SP ; SET THE USER STACK TO VA 100
1312 006104 005037 177776 CLR 2#PS
```

1313	006110	012777	077406	172756	MOV	#77406, #KPDRO	;MAP KERNEL, AND USER TO BANK 0,4K,RM
1314	006116	012777	077406	172710	MOV	#77406, #UPDRO	
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	006138	016701	172662		MOV	SRO, R1	:REFERENCE SRO 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

DFKTR.A MACY11 27(732) 09-SEP-76 17:12 PAGE 28
DFKTR.A.P11

1321	006162	005277	172636			INC	2SR0			
1322	006166	000004				IOT				:TURN ON MEMORY MANAG.
1323	006170	000240				NOP				:SHOULD USE STACK IN KERNEL ADDRESS SPACE
1324	006172	005011				NOP				
1325	006174	012737	006226	000020		KRET34: CLR	2R1			:TURN OFF MEMORY MANAG.
1326	006202	012737	140000	000022		MOV	2URET34,220			:SETUP FOR IOT TO USER
1327	006210	012737	140000	177776		MOV	2140000,222			
1328	006216	005277	172602			MOV	2140000,22PS			
1329	006222	000004				INC	2SR0			:TURN ON MEMORY MANAG.
1330	006224	000240				IOT				:SHOULD USE STACK IN USER SPACE
1331	006226	005011				NOP				
1332	006230	022737	006170	000474		URET34: CLR	2R1			:TURN OFF MEMORY MANAG.
1333	006236	001401				CMP	2KRET34-2,22474			
1334	006240	104006				BEQ	.+4			
1335	006242	022737	000000	000476		HLT				:KERNEL STACK CONTENTS WRONG. PC NOT WHERE IT
						CMP	20,22476			:SHOULD HAVE BEEN PUSHED OR

1336	006250	001401		
1337	006252	104006		
1338	006254	022737	006224	000074
1339	006256	001401		
1340	006258	104006		
1341	006256	022737	140000	000076
1342	006274	001401		
1343	006276	104006		
1344				
1345				
1346	006300	012737	000076	000074
1347	006306	005037	000076	
1348	006312	012737	000476	000474
1349	006320	005037	000476	
1350	006324	012706	001000	
1351				
1352				
1353				
1354				
1355				
1356				
1357				
1358	006330	104400		
1359	006332	012706	001000	
1360	006336	004767	010042	
1361	006342	000030		
1362	006344	104306		
1363	006346	005077	172452	
1364	006352	004767	007036	
1365	006356	012777	000001	172470
1366	006358	004767	010100	
1367	006360	012777	007600	172474
1368	006376	016701	172422	
1369	006402	012737	140000	177776
1370	006410	012706	000400	
1371	006414	00337	177776	
1372	006420	012706	001000	
1373	006424	012737	006564	000130
1374	006432	012737	005602	000030
1375	006440	00337	000132	
1376	006444	005037	000032	
1377	006450	012737	140000	177776
1378	006456	005277	172342	
1379	006462	000000		
1380	006464	000000		
1381	006466	000000		
1382	006470	000000		
1383	006472	000000		
1384	006474	000000		
1385	006476	000000		
1386	006500	000000		
1387	006502	000000		
1388	006504	000000		
1389	006506	000000		
1390	006510	000000		
1391	006512	000000		

BEG	.+4	: VALUE WRONG
HLT		: KERNEL STACK WRONG-TRAP STATUS NOT
CHP	0URET34-2,0074	: NOT WHERE IT SHOULD HAVE BEEN PUSHED
BEG	.+4	: OR VALUE WRONG
HLT		: USER STACK WRONG-PC NOT WHERE
CHP	0140000,0076	: IT SHOULD HAVE BEEN PUSHED
BEG	.+4	: OR VALUE WRONG
HLT		: USER STACK WRONG-TRAP STATUS
		: NOT WHERE IT SHOULD HAVE BEEN
		: PUSHED OR VALUE WRONG
		: REINITIALIZE LOCATIONS CHECKED
MOV	076,0074	
CLR	0076	
MOV	0476,00474	
CLR	00476	
MOV	0KSTACK,SP	
: SHOW THAT TRAP ENT. AND INTERRUPTS TAKE VECTORS FROM KERNEL		
: IRREGARDLESS OF THE MODE AT THE TIME OF THE TRAP SEQUENCE		
: ALSO SHOW THAT 000-ADDRESS TRAP (AN "INTERNAL"		
: TRAP) TAKES ITS VECTOR FROM KERNEL		
: NOTE THAT IF DUAL ADDRESSING OCCURS, THE ERROR		
: ADDRESS WILL BE USED (THE 0 OVERRIDES THE 1)		
TEST30: SCOPE		
MOV	0KSTACK,SP	: INITIALIZE KERNEL STACK POINTER
JSR	PC,ORDER	: CHECK TEST SEQUENCE + INIT SR0
30		: TEST NUMBER
HLT		: TEST EXECUTED OUT OF SEQUENCE
CLR	0SR0	
JSR	X7,0KALL	: MAP ALL PAR/PDR PAIR'S RN, 4K, BANK 0
MOV	01,0UPAR0	: OFFSET USER 0 1 PAGE
JSR	PC,KERN7	: MAP KERNEL PAR/PDR 7 TO EXT BANK
MOV	07600,0UPAR7	: MAP USER 7 TO THE EXTERNAL BANK
MOV	SR0,R1	: SETUP R1 TO REFERENCE SR0
MOV	0140000,00PS	: SETUP USER STACK
MOV	0KSTACK,SP	
MOV	00K358,00130	: SETUP THE KERNEL STACK POINTER
MOV	00K358,0030	: SETUP FAILURE RETURN
CLR	00132	: SETUP SUCCESS RETURN
CLR	0032	
MOV	0140000,00PS	: SET MODE TO USER
INC	0SR0	: TURN ON MEMORY MANAG.

```

1398 006514 000000
1399 006516 000000
1400 006520 000000
1401 006522 000000
1402 006524 000000
1403 006526 000000
1404 006528 000000
1405 006530 000000
1406 006532 000000
1407 006534 000000
1408 006536 000000
1409 006538 000000
1410 006540 000000
1411 006542 000000
1412 006544 104000
1413 006546 022626
1414 006548 005011
1415 006550 012737
1416 006552 104006
1417 006554 000402
1418 006556 022626
1419 006558 005011
1420 006560 012737
1421 006562 012737
1422 006564 012737
1423 006566 005037
1424 006568 012737
1425 006570 012737
1426 006572 005037
1427 006574 005037
1428 006576 012737
1429 006578 140000
1430 006580 177776
1431 006582 005277
1432 006584 000000
1433 006586 000000
1434 006588 000000
1435 006590 000000
1436 006592 000000
1437 006594 000000
1438 006596 000000
1439 006598 000000
1440 006600 000000
1441 006602 000000
1442 006604 000000
1443 006606 000000
1444 006608 000000
1445 006610 000000
1446 006612 000000
1447 006614 000000
1448 006616 000000
1449 006618 000000
1450 006620 000000
1451 006622 000000
1452 006624 000000
1453 006626 000000
1454 006628 000000
1455 006630 000000
1456 006632 000000
1457 006634 000000
1458 006636 000000

```

```

016334 000030
016334 000030
000340 000032
000132 000130
000132
006774 000120
007004 000020
000122
000022
140000 177776
172132

```

```

NG35B:
OK35B:
IOT35:

```

```

ENT
CMP (SP)+,(SP)+
CLR @R1
MOV @EMTSRV,@#30
BR T
CMP (SP)+,(SP)+
CLR @R1
MOV @EMTSRV,@#30
MOV @340,@#32
MOV @132,@#130
CLR @#132
MOV @NG35C,@#120
MOV @OK35C,@#20
CLR @#122
CLR @#22
MOV @140000,@#PS
INC @SR0

```

```

; SHOULD PICK UP RETURN ADDRESS FROM KERNEL
; RESTORE STACK POINTER
; TURN OFF MEMORY MANAG.

; ENT VECTOR DIDN'T GO THRU KERNEL

; RESTORE STACK POINTER
; TURN OFF MEMORY MANAG.
; RESTORE ENT SERVICE POINTER

; SETUP IOT FAILURE RETURN
; SETUP IOT SUCCESS RETURN

; SET MODE TO USER
; TURN ON MEMORY MANAG.

```

```

1446 006740 000000
1447 006742 000000
1448 006744 000000
1449 006746 000000
1450 006748 000000
1451 006750 000000
1452 006752 000000
1453 006754 000000
1454 006756 000000
1455 006758 000000
1456 006760 000000
1457 006762 000000
1458 006764 000000
1459 006766 000000
1460 006770 000000
1461 006772 000004
1462 006774 022626
1463 006776 005011
1464 007000 104006
1465 007002 000402
1466 007004 022626
1467 007006 005011
1468 007010 012737 000022 000020
1469 007016 005037 000022
1470 007022 012737 000122 000120
1471 007030 005037 000122
1472 007034 012737 007216 000164
1473 007042 012737 007232 000064
1474 007050 005037 000166
1475 007054 005037 000066
1476 007060 012737 140000 177776
1477 007066 005277 171732
1478 007072 000000
1479 007074 000000
1480 007076 000000
1481 007100 000000
1482 007102 000000
1483 007104 000000
1484 007106 000000
1485 007110 000000
1486 007112 000000
1487 007114 000000
1488 007116 000000
1489 007120 000000
1490 007122 000000
1491 007124 000000
1492 007126 000000
1493 007130 000000
1494 007132 000000
1495 007134 000000
1496 007136 000000
1497 007140 000000
1498 007142 000000
1499 007144 000000
1500 007146 000000
1501 007150 000000
1502 007152 000000
1503 007154 000000

```

```

NG35C:
OK35C:
INT35:

```

```

TOT
CMP (SP)+,(SP)+
CLR @R1
HLT
BR INT35
CMP (SP)+,(SP)+
CLR @R1
MOV @22,@#20
CLR @#22
MOV @122,@#120
CLR @#122
MOV @NG350,@#164
MOV @OK350,@#64
CLR @#166
CLR @#66
MOV @140000,@#PS
INC @SR0

```

```

; SHOULD PICK UP RETURN ADDRESS FROM KERNEL
; RESTORE STACK POINTER
; TURN OFF MEMORY MANAG.
; TRAP VECTOR DIDN'T GO THRU KERNEL

; RESTORE STACK POINTER
; TURN OFF MEMORY MANAG.

; SETUP TTY FAILURE RETURN
; SETUP TTY SUCCESS RETURN

; SET MODE TO USER
; TURN ON MEMORY MANAG.

```


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	000AD		
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	000AD		
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	000AD:	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	@SRO		; TURN ON MEMORY MANAG.
1536	007332	000000							
1537	007334	000000							
1538	007336	000000							
1539	007340	000000							
1540	007342	000000							
1541	007344	000000							
1542	007346	000000							
1543	007350	000000							
1544	007352	000000							
1545	007354	000000							
1546	007356	000000							
1547	007360	000000							
1548	007362	000000							
1549	007364	000000							
1550	007366	000000							
1551	007370	000000							
1552	007372	000000							
1553	007374	000000							
1554	007376	000000							
1555	007400	000000							
1556	007402	000000							
1557	007404	000000							
1558	007406	000000							
1559	007410	000000							

1616	007652	042777	000001	171144	INT36A:	BIC	#1, %SR0	;TURN OFF MEMORY MANAG.
1617	007650	022777	100002	171136		CMP	#100002, %SR0	;CHECK SR0
1618	007656	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 RETL N ADDRESS
1628	007724	005077	171074			CLR	%SR0	;CLEAR NR ERROR BIT-SHOULD
1629								;UNLOCK ENR OR 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 %P 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/POR 7 TO EXT BANK
1663	010062	012777	077406	171004		MOV	#77406, %KPOR0	;MAP KERNEL 0 RW, RK, BANK0
1664	010070	012777	017406	171000		MOV	#17406, %KPOR1	;MAP KERNEL 1 PL, LK, 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	000077	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

K03

DFKTR.A MACY11 27(732) 09-SEP-76 17:12 PAGE 36
DFKTR.A.P11

Address	Hex	Hex	Hex	Label	Instruction	Comments
1728	010404	104006			HLT	: TEST EXECUTED OUT OF SEQUENCE
1729	010406	004767	004756		JSR	: CLEAR ALL MEMORY MANAG. REGISTERS
1730	010412	004767	006052		JSR	: MAP KERNEL PAR/PDR 7 TO EXT BANK
1731	010416	012777	077406	170450	MOV	: MAP KERNEL 0 RW,RK,BANK0
1732	010424	012777	077402	170444	MOV	: MAP KERNEL 1 ACC 4 K,BANK0
1733	010432	012777	010466	170524	MOV	: SETUP RETURN VECTOR
1734	010440	005077	170522		CLR	
1735	010444	005077	170354		INC	
1736	010450	013737	037776	ADR40:	MOV	: TURN ON MEMORY MANAG.
1737	010456	005077	170342		CLR	: REFERENCE KERNEL 1 - 1ST ABORT
1738	010462	104006			HLT	: TURN OFF MEMORY MANAG.
1739	010464	005110			BR	: REFERENCE TO KERNEL 1
1740	010466	042777	000001	170330	INT40:	: DIDN'T ABORT
1741	010474	022777	020002	170322	BIC	: TURN OFF MEMORY MANAG.
1742	010502	001401			CHP	: CHECK SR0
1743	010504	104006			BEQ	
1744	010506	012777	010542	170450	HLT	: SR0 INCORRECT AFTER ACC ABORT
1745	010514	022626			MOV	: SETUP NEW RETURN VECTOR
1746	010516	012732	037776		CHP	: RESTORE STACK POINTER
1747	010522	052777	000001	170274	MOV	: SETUP R2 TO REFERENCE KERNEL 1
1748	010530	012242			BIS	: TURN ON MEMORY MANAG.
1749	010532	005077	170266	ADR40A:	MOV	: REFERENCE KERNEL 1 - 2ND ABORT
1750	010536	104006			CLR	: TURN OFF MEMORY MANAG.
1751	010540	004622			HLT	: 2ND REFERENCE TO KERNEL 1
1752	010542	042777	000001	170254	INT40A:	: DIDN'T ABORT
1753	010548	022777	020002	170246	BIC	: TURN OFF MEMORY MANAG.
1754	010556	001401			CHP	: CHECK SR0
1755	010560	104006			BEQ	
1756	010562	022777	010450	170242	HLT	: SR0 INCORRECT AFTER 2ND ACC ABORT
1757	010570	001401			CHP	: CHECK SR2
1758	010572	104006			BEQ	
1759	010574	021627	010532		HLT	: SR2 DOESN'T CONTAIN VALUE FROM 1ST ABORT
1760	010600	001401			CHP	: CHECK ADDRESS PUSHED ON STACK
1761	010602	104006			BEQ	
1762	010604	022626			HLT	: INCORRECT ADDRESS ON STACK
1763	010606	012777	010642	170350	CHP	: RESTORE STACK POINTER
1764	010614	005077	170204		MOV	: CHANGE RETURN ADDRESS
1765					CLR	: CLEAR ACC ERROR BIT-SHOULD
1766	010620	012702	037776		MOV	: "UNLOCK" ERROR TRACKING
1767	010624	005277	170174		INC	: SETUP R2 TO REFERENCE KERNEL 1
1768	010630	012242			ADR40B:	: TURN ON MEMORY MANAG.
1769	010632	005077	170166	ADR40C:	MOV	: 3RD ACC REFERENCE, ERROR BIT WAS CLEARED
1770	010636	104006			CLR	: TURN OFF MEMORY MANAG.
1771	010640	004422			HLT	: 3RD REFERENCE TO KERNEL 1
1772	010642	042777	000001	170154	INT40B:	: DIDN'T ABORT
1773	010650	022777	020002	170146	BIC	: TURN OFF MEMORY MANAG.
1774	010656	001401			CHP	: CHECK SR0
1775	010660	104006			BEQ	
1776	010662	022777	010630	170142	HLT	: SR0 INCORRECT
1777	010670	001401			CHP	: CHECK SR2
1778	010672	104006			BEQ	
1779					HLT	: SR2 INCORRECT - SHOULD CONTAIN
1780	010674	022716	010632		CHP	: LAST FETCH ADDRESS BEFORE ABORT
1781	010700	001401			BEQ	: CHECK STACK
1782	010702	104006			HLT	
1783	010704	022626			CHP	: PC ON STACK INCORRECT
						: RESTORE STACK POINTER

L03

```

1784 010706 005077 170112      DONE40: CLR      JSRO      ;CLEAR ERROR BIT
1785 010712 005077 170250      CLR      ;CHANGE TRAP RETURN TO CAUSE A HALT
1786 010716 016777 170244 170240  MOV      KTSTA,2KTVEC ;ON A FALSE INTERRUPT
1787
1788
1789 010724 104400      ;SHOW THAT INIT CLEARS SRO(13-15)
1790 010726 012706 001000  TEST34: SCOPE
1791 010732 004767 005446      MOV      *KSTACK,SP ;INITIALIZE KERNEL STACK POINTER
1792 010736 000034      JSR      PC,ORDER    ;CHECK TEST SEQUENCE + INIT SRO
1793 010740 104006      34        ;TEST NUMBER
1794 010742 112777 000340 170056  HLT      ;TEST EXECUTED OUT OF SEQUENCE
1795 010750 122777 000340 170050  MOVB     #340,JSROH  ;SET SRO BITS 13-15
1796 010756 001401      CMPS     #340,JSROH ;MAKE SURE THEY SET CORRECTLY
1797 010760 104006      BEQ     .+4
1798 010762 000005      HLT      ;SRO INCORRECT (HIGH BYTE)
1799 010764 122777 000000 170034  RESET    ;ISSUE INIT
1800 010772 001401      CMPS     #0,JSROH   ;CHECK SRO HIGH BYTE
1801 010774 104006      BEQ     .+4
1802 010776 012767 000010 004652  HLT      ;SRO INCORRECT AFTER INIT
1803      MOV     #10,ICOUNT ;DROP ITERATION COUNT
1804
1805 011004 104400      ;SHOW THAT INIT CLEARS SRO AFTER ABORT
1806 011006 012706 001000  TEST35: SCOPE
1807 011012 004767 005366      MOV      *KSTACK,SP ;INITIALIZE KERNEL STACK POINTER
1808 011016 000035      JSR      PC,ORDER    ;CHECK TEST SEQUENCE + INIT SRO
1809 011020 104006      35        ;TEST NUMBER
1810 011022 004767 004366  JSR      ;TEST EXECUTED OUT OF SEQUENCE
1811 011026 012777 000416 170040  MOV      X7,RWALL    ;MAP ALL PAR/PDR PAIR'S 4K,RW BANK 0
1812      ;MAP KERNEL 0 RW,4K LESS 1 PAGE
1813 011034 004767 005430      JSR      ;DOWN (100-17776 RW)
1814 011040 012777 077400 170030  MOV      PC,KERN7    ;MAP KERNEL PAR/PDR 7 TO EXT BANK
1815 011046 012777 011102 170110  MOV      #77400,2KPOR1 ;MAP KERNEL PAGE 1 NR
1816 011054 005077 170106      CLR      #RET2,2KTVEC ;SETUP ABORT RETURN
1817 011060 012746 000020      MOV      #20,-(SP)   ;SET T BIT IN STATUS ON STACK
1818 011064 012746 011076      MOV      #AOR2,-(SP) ;SETUP ADDRESS ON STACK
1819 011070 005277 167730      INC      JSRO        ;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      AOR2:   HALT        ;NO PL ABORT OCCURRED
1825 011100 000412      BR      DONE2
1826 011102 022777 040001 167714  RET2:   CMP      #40001,JSRO ;CHECK SRO
1827 011110 001401      BEQ     .+4
1828 011112 104006      HLT      ;SRO INCORRECT - SHOULD SHOW
1829      ;REFERENCE TO KERNEL 0
1830
1831 011114 000005      RESET    ;AND PL ABORT SHOULD BE SET
1832 011116 005777 167702      TST     JSRO        ;ISSUE INIT - SHOULD CLEAR SRO
1833 011122 001401      BEQ     .+4        ;CHECK SRO
1834 011124 104006      HLT      ;SRO INCORRECT AFTER INIT
1835 011126 005077 167672      DONE2:  CLR      JSRO    ;REINITIALIZE SRO
1836 011132 016777 170030 170024  MOV      KTSTA,2KTVEC
1837 011140 012737 000016 000014  MOV      #16,2B14   ;RESTORE T-BIT TRAP CATCHER
1838
1839      ;SHOW THAT INIT CLEARS SRO(0-3,5-6)

```

M03

DFKTR.A MACY11 27(732) 09-SEP-76 17:12 PAGE 38
DFKTR.A.P11

```

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 SRO
1845 011160 000036              36          ;TEST NUMBER
1846 011162 104006              HLT          ;TEST EXECUTED OUT OF SEQUENCE
1847 011164 004767 004224      JSR      %7, R%ALL      ;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      @SRO          ;TURN ON MEMORY MANAG.
1856 011232 005737 160000      TST      @#160000      ;REFERENCE PAGE 7
1857 011236 000777              BP          ;NO ABORT ON NR REFERENCE
1858 011240 022777 100157 167556  RET3:    CMP      #100157, @SRO ;CHECK SRO
1859 011246 001401              BEQ      .+4
1860 011250 104006              HLT          ;SRO INCORRECT - SHOULD HAVE TRACKED
1861                                     ;NR REFERENCE TO USER 7
1862 011252 000005              RESET
1863 011254 005777 167544      TST      @SRO          ;ISSUE INIT
1864 011260 001401              BEQ      .+4          ;CHECK SRO
1865 011262 104006              HLT          ;SRO INCORRECT AFTER INIT
1866 011264 005077 167534      CLR      @SRO
1867 011270 012767 000010 004360  MOV      #10, I%COUNT ;DROP ITERATION COUNT
1868 011276 016777 167664 167660  MOV      KTSTA, @KTVEC
1869
1870                                     ;SHOW THAT BYTE ADDRESSING OF SRO 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 SRO
1874 011316 000037              37          ;TEST NUMBER
1875 011320 104006              HLT          ;TEST EXECUTED OUT OF SEQUENCE
1876 011322 004767 004066      JSR      %7, R%ALL      ;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, @SRO ;TURN ON MEMORY MANAG. AND SET ERROR FLAGS
1879 011340 105077 167460      CLR%      @SRO          ;DATOB (LOW) TO SRO
1880 011344 032777 160000 167452  BIT      #160000, @SRO ;CHECK SRO
1881 011352 001001              BNE      .+4
1882 011354 104006              HLT          ;SRO INCORRECT AFTER DATOB
1883 011356 012777 160001 167440  MOV      #160001, @SRO ;DATOB (HIGH) TO SRO
1884 011364 105077 167436      CLR%      @SRO%        ;CHECK SRO
1885 011370 022777 000017 167426  CMP      #17, @SRO
1886 011376 001401              BEQ      .+4
1887 011400 104006              HLT          ;SRO INCORRECT AFTER DATOB
1888 011402 005077 167416      CLR      @SRO
1889
1890                                     ;SHOW THAT SRO (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 SRO 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 SRO

```

```

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
1900 011434 012777 011512 167522  MOV                #RET5,%KTVEC    ; MAP KERNEL PAR/PDR 7 TO EXT BANK
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          MOV                #77400,%R1    ; MAKE USER NR
1907 011470 012737 140000 177776  MOV                #140000,%#PS  ; ENTER USER MODE
1908 011476 005277 167322          INC                %SRO
1909 011502 005712                TST                %R2
1910 011504 000777                BR                 .
1911 011506 000005                RESET
1912 011510 000423                BR                 DONES
1913 011512 017705 167306          MOV                %SRO,%R5    ; SAVE CONTENTS OF SRO
1914 011516 005077 167302          CLR                %SRO        ; TURN OFF MEMORY MANAG.
1915 011522 020503                CMP                R5,R3        ; CHECK SAVED CONTENTS OF SRO
1916 011524 001401                BEQ                .+4
1917 011526 104006                HLT
1918 011530 020167 167300          CMP                R1,UPDR0    ; SRO INCORRECT
1919 011534 001002                BNE                LOPSA        ; IS USER 0 UNDER TEST
1920 011536 012711 077406          MOV                #77406,%R1  ; NO, CONTINUE
1921 011542 022626                CMP                (R6)+,(R6)+ ; MAKE USER 0 RESIDENT
1922 011544 005721                TST                (R1)+
1923 011546 062703 000002          ADD                #2,R3
1924 011552 062702 020000          ADD                #20000,R2
1925 011556 077436                SOB                R4,LOPSA
1926 011560 016777 167402 167376  MOV                %KTSTA,%KTVEC ; DONES:
1927 011566 005077 167374          CLR                %KTSTA
1928
1929                ; SHOW THAT SRO <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 SRO AFTER AN ABORT
1932 011572 104400                TEST41: SCOPE
1933 011574 012706 001000          MOV                #KSTACK,SP  ; INITIALIZE KERNEL STACK POINTER
1934 011603 004767 004600          JSR                PC,ORDER    ; CHECK TEST SEQUENCE + INIT SRO
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                %SRO        ; TURN ON MEMORY MANAG.
1943 011646 005737 020000          TST                #20000     ; REFERENCE PAGE 1 (NR)
1944 011652 005077 167146          CLR                %SRO        ; TURN OFF MEMORY MANAG.
1945 011656 104006                HLT                ; NR REFERENCE DIDN'T ABORT
1946 011660 000436                BR                 DONE7
1947 011662 017701 167136          MOV                %SRO,R1    ; SAVE SRO CONTENTS IN R1
1948 011666 005077 167132          CLR                %SRO        ; TURN OFF MEMORY MANAG.
1949 011672 022701 100003          CMP                #100003,R1  ; CHECK SAVED CONTENTS OF SRO
1950 011676 001401                BEQ                .+4
1951 011700 104006                HLT
; SRO INCORRECT SHOULD SHOW NR ERR, KERNEL PAGE 1

```


1953	011702	012777	011736	167254		MOV	RET7C, &KTVEC	: SETUP NEXT ABORT RETURN
1954	011710	012737	140000	177776		MOV	#140000, &@PS	: CHANGE MODE TO USER
1954	011716	005277	167102			INC	&SRO	: TURN ON MEMORY MANAG.
1955	011722	005737	020000			TST	&@20000	: REFERENCE USER PAGE 1 (NR)
1956	011726	005077	167072			CLR	&SRO	: TURN OFF MEMORY MANAG.
1957	011732	104006				HLT		: NR REFERENCE DIDN'T ABORT
1958	011734	000410				BR	DONE7	
1959	011736	017701	167062		RET7C:	MOV	&SRO, R1	: SAVE CONTENTS OF SRO
1960	011742	005077	167056			CLR	&SRO	: TURN OFF MEMORY MANAG.
1961	011746	022701	100143			CMP	#100143, R1	: CHECK SAVED CONTENTS OF SRO
1962	011752	001401				BEQ	.+4	
1963	011754	104006				HLT		: SRO 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 SRO (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 SRO
1972	011776	000042				42		: TEST NUMBER
1973	012000	104006				HLT		: TEST EXECUTED OUT OF SEQUENCE
1974	012002	004767	003406			JSR	X7, R#ALL	: SET ALL PAR/PDR PAIRS RW, WK, 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	&SRO	: TURN ON MEMORY MANAG.
1978	012026	042777	000001	166770		BIC	#1, &SRO	: TURN OFF MEMORY MANAG.
1979	012034	005037	177776			CLR	&@PS	: CHANGE TO KERNEL MODE
1980	012040	022777	000156	166756		CMP	#156, &SRO	: CHECK SRO
1981	012046	001401				BEQ	.+4	
1982	012050	104006				HLT		: SRO 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	&SRO	
1989								
1990								: SHOW THAT IF AN INSTRUCTION IS COMPLETED BEFORE A MEMORY MANAGEMENT FAULT
1991								: OCCURS, SR2 WILL CONTAIN THE ADDRESS OF LAST 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	PC, ORDER	: CHECK TEST SEQUENCE + INIT SRO
1999	012070	000043				43		: TEST NUMBER
2000	012072	104006				HLT		: TEST EXECUTED OUT OF SEQUENCE
2001	012074	004767	003314			JSR	X7, R#ALL	: INITIALIZE ALL PAGES RW, WK, BANK 0
2002	012100	012777	000416	166766		MOV	#416, &KPOR0	: 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

C04

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 41
DFKTR.A.P11

2008	012130	012746	012136			MOV	R,+6,-(SP)		
2009	012134	000006				RTT			:SET T-BIT VIA RTT
2010	012136	012777	000001	166660	ADR11:	MOV	R1,RSRO		:TURN ON MEMORY MANAG. - SHOULD
2011									:ATTEMPT TO TRACE TRAP AT END OF
2012									:INSTRUCTION - SHOULD GET A PAGE
2013									:LENGTH ERROR ON THAT ATTEMPT
2014	012144	000000				HALT			:NO PAGE LENGTH ERROR ON TRACE TRAP
2015	012146	000415				BR	CONT11		
2016	012150	042777	000001	166646	RET11:	BIC	R1,RSRO		:TURN OFF MEMORY MANAG.
2017	012156	022777	040000	166640		CHP	R40000,RSRO		:CK SRO
2018	012164	001401				BEQ	.+4		
2019	012166	104006				HLT			:SRO INCORRECT - PL FAULT, KERNEL 0 REFERENCE COMPLETED
2020	012170	022777	012136	166634		CHP	RADW11,RSR2		: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	RSRO		:REINITIALIZE SRO
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	RKSTACK,SP		:INITIALIZE KERNEL STACK POINTER
2031	012222	004767	004156			JSR	PC,ORDER		:CHECK TEST SEQUENCE + INIT SRO
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/POR PAIRS TO RW,4K, BANK 0
2035	012236	012777	077402	166634		MOV	R77402,PKPDR2		:SET KERNEL PAR/POR PAIR 2 RRO,4K
2036	012244	004767	004220			JSR	PC,KERY7		:MAP KERNEL PAR/POR 7 TO EXT BANK
2037	012250	012777	012312	166706		MOV	RRET13A,KTVEC		:SETUP MEMORY MANAGEMENT ABORT RETURN
2038	012256	005077	166704			CLR	KTSTA		
2039	012262	005277	166536			INC	RSRO		:TURN ON MEMORY MANAG.
2040	012266	012777	160001	166530		MOV	R160001,RSRO		:SET ABORT ERROR BITS
2041	012274	013737	007000	047000		MOV	R07000,R047000		:WRITE KERNEL PAR/POR PAIR 2 (RRO)-SHOULD TRAP
2042	012302	005077	166516			CLR	RSRO		
2043	012306	104006				HLT			:NO TRAP OCCURRED
2044	012310	000416				BR	DONE13		
2045	012312	022626			RET13A:	CHP	(SP)+(SP)+		:RESTORE THE STACK POINTER
2046	012314	017701	166504			MOV	RSRO,R1		:SAVE CONTENTS OF SRO
2047	012320	005077	166500			CLR	RSRO		:TURN OFF MEMORY MANAG.
2048	012324	022701	160017			CHP	R160017,R1		
2049	012330	001401				BEQ	.+4		
2050	012332	104006				HLT			:SAVED CONTENTS OF SRO INCORRECT
2051	012334	022777	077402	166536		CHP	R77402,PKPDR2		:CHECK THE POR CORRESPONDING TO THE TRAP REFERENCE
2052	012342	001401				BEQ	.+4		
2053	012344	104006				HLT			:THE POR CORRESPONDING TO THE TRAP REFERENCE IS INCORRECT
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	RSRO		:REINITIALIZE SRO
2057									
2058									:SHOW THAT MEMORY MANAGEMENT WILL NOT TRAP ON AN INTERNAL REFERENCE
2059	012360	104400							TEST45: SCOPE
2060	012362	012706	001000			MOV	RKSTACK,SP		:INITIALIZE KERNEL STACK POINTER
2061	012366	004767	004012			JSR	PC,ORDER		:CHECK TEST SEQUENCE + INIT SRO
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	@KTSTA	
2068	012420	005277	166400		INC	@SRO	:TURN ON MEMORY MANAG.
2069	012424	005777	166374		TST	@SRO	:TRAP REFERENCE TO A MEMORY MANAG. REGISTER
2070	012430	005077	166370		CLR	@SRO	
2071	012436	022777	077406	166450	CMP	@77406,@KPOR7	
2072	012442	001401			BEG	.+4	
2073	012448	104006			HLT		
2074	012454	000404			BR	DONE16	
2075	012460	042777	000001	166346	RET16:	BIC	@1,@SRO
2076	012466	104006			HLT		
2077	012472	005077	166340		DONE16:	CLR	@SRO
2078	012478	016777	166476	166472	MOV	KTSTA,@KTVEC	
2079							
2080							
2081							
2082							
2083							
2084	012472	104000					
2085	012474	012706	001000		MOV	@KSTACK,SP	:INITIALIZE KERNEL STACK POINTER
2086	012500	004767	003700		JSR	PC,ORDER	:CHECK TEST SEQUENCE + INIT SRO
2087	012504	000046			46		:TEST NLT
2088	012506	104006			HLT		:TEST EXCLUDED 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 FILTER 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	@SRO
2098	012546	010277	166324		MOV	R2,@KPOR1	:TURN ON MEMORY MANAG.
2099	012552	005727	020000		TST	@20000	:SET KERNEL PAR/PDR PAIR 1 TO NEW PAGE LENGTH
2100	012556	005711			TST	@R1	:READ LOWER BOUNDARY-SHOULDN'T ABORT
2101							:READ UPPER ALLOWED BOUNDARY-SHOULDN'T
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	@SRO	:TURN MEMORY MANAG. OFF
2109	012604	104306			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,@SRO
2112	012616	022626			CMP	(SP)+,(SP)+	:TURN OFF MEMORY MANAG.
2113	012620	104306			HLT		:RESTORE STACK POINTER
2114	012622	005077	166176		CLR	@SRO	:REFERENCE WITHIN ALLOWED AREA
2115	012626	000416			BR	CONT23	:CLEAR ERROR BITS
2116	012630	022626			RET23B:	CMP	(SP)+,(SP)+
2117	012632	017703	166166		MOV	@SRO,R3	:CAUSED A TRAP OR ABORT
2118	012636	005077	166162		CLR	@SRO	:RESTORE STACK POINTER
2119	012642	022703	040003		CMP	@40003,R3	:SAVE CURRENT SRO
							:TURN OFF MEMORY MANAG.
							:CK SAVED SRO

:TEST PAGE LENGTH ERROR CHECKING (EXPAND DOWN NOT SET)
 :KERNEL PAR/PDR PAIR 1 IS USED WITH ALL PAGE LENGTH VALUES
 :SHOW THAT REFERENCES TO BOTH BOUNDARIES OF THE ALLOWED AREA DON'T TRAP OR ABORT
 :SHOW THAT A REFERENCE TO THE FIRST WORD BEYOND THE ALLOWABLE AREA DOES TRAP

TEST46: SCOPE
 :INITIALIZE KERNEL STACK POINTER
 :CHECK TEST SEQUENCE + INIT SRO
 :TEST NLT
 :TEST EXCLUDED OUT OF SEQUENCE
 :INITIALIZE ALL PAR/PDR PAIRS TO RW 4K, BANK 0
 :MAP KERNEL PAR/PDR 7 TO EXT BANK
 :R2 CONTAINS VALUE TO BE LOADED IN THE
 :PDR BEING CHECKED (INCLUDING PLF)
 :R1 IS USED TO FILTER THE TOP ADDRESS
 :WITHIN THE ALLOWED AREA
 :SETUP ABORT RETURN IN CASE REFERENCE
 :WITHIN ALLOWED AREA ABORTS
 :TURN ON MEMORY MANAG.
 :SET KERNEL PAR/PDR PAIR 1 TO NEW PAGE LENGTH
 :READ LOWER BOUNDARY-SHOULDN'T ABORT
 :READ UPPER ALLOWED BOUNDARY-SHOULDN'T
 :ABORT
 :SETUP ABORT RETURN
 :CHECK FOR DONE (TO AVOID REFERENCING
 :NEXT PAR/PDR PAIR)
 :EXIT LOOP IF DONE
 :REFERENCE OUTSIDE ALLOWED AREA -
 :SHOULD ABORT
 :TURN MEMORY MANAG. OFF
 :NO ABORT OCCURRED ON A REFERENCE
 :OUTSIDE THE ALLOWED PAGE LENGTH
 :TURN OFF MEMORY MANAG.
 :RESTORE STACK POINTER
 :REFERENCE WITHIN ALLOWED AREA
 :CLEAR ERROR BITS
 :CAUSED A TRAP OR ABORT
 :RESTORE STACK POINTER
 :SAVE CURRENT SRO
 :TURN OFF MEMORY MANAG.
 :CK SAVED SRO

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 43
DFKTR.A.P11

2120	012646	001401		BEQ	.+4		
2121	012650	104006		HLT			: CONTENTS OF SRO INCORRECT AFTER
2122							: PAGE LENGTH ERROR
2123	012652	022777	000002	CMP	#2, #SRO	166144	: CHECK SRO TO BE SURE PL BIT CLEARED
2124	012660	001401		BEQ	.+4		
2125	012662	104006		HLT			: SRO 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	#SRO	166254	: TURN OFF MEMORY MANAG.
2133	012702	016777	166260	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 PAIR 1 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 FOR SRO = 0 + INIT SRO
2144	012726	000047		47			: TEST MEMORY
2145	012730	104006		HLT			: TEST EXECUTED OUT OF SEQUENCE
2146	012732	004767	002456	JSR	#7, #R1ALL		: INITIALIZE ALL PAR/PDR PAIRS TO RM, #K, BANK 0
2147	012736	004767	003526	JSR	PC, KERN7		: MAP KERNEL PAIR #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	MOV	#RET24A, #KTVEC	166204	: SETUP ABORT RETURN IN CASE REFERENCE
2153	012760	005077	166202	CLR	#KTSTA		: WITHIN ALLOWED AREA ABORTS
2154	012764	005277	166034	LOOP24: INC	#SRO		: TURN ON MEMORY MANAG.
2155	012770	010277	166102	MOV	R2, #KPDRI		: 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	MOV	#RET24B, #KTVEC	166154	: 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	#SRO		: 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	#SRO		: 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	#SRO, R3		: SAVE CURRENT SRO
2174	013052	005077	165746	CLR	#SRO		: TURN OFF MEMORY MANAG.
2175	013056	022703	040003	CMP	#40003, R3		: CK SAVED SRO

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			TESTS0: 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			SO			: 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/POR 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,KTSTOR1		: 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			CHP	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							
2244	013336	104400					;SHOW THAT THE W BIT DOESN'T SET IF THE MEMORY MANAG. IS OFF
2245	013340	012706	001000		TESTS1: SCOPE		
2246	013344	004767	003034		MOV	@KSTACK, SP	;INITIALIZE KERNEL STACK POINTER
2247	013350	000051			JSR	PC, ORDER	;CHECK TEST SEQUENCE + INIT SR0
2248	013352	104306			S1		;TEST NUMBER
2249	013354	012767	002000	002274	HLT		;TEST EXECUTED OUT OF SEQUENCE
2250	013362	004767	002002		MOV	@2000, ICOUNT	;RESTORE ITERATION COUNT
2251	013366	013737	010000	010000	JSR	X7, CLRALL	;CLEAR ALL MEMORY MANAG. REGISTERS
2252	013374	005777	165474		MOV	@10000, @10000	;WRITE BANK 0
2253	013400	001401			TST	@KPDOR	
2254	013402	104006			BEQ	.+4	
2255					HLT		;W BIT SET OR ANOTHER BIT INCORRECT
2256							;IN KERNEL 0 PDR
2257							
2258							
2259	013404	104400					;SHOW THAT THE W BIT IS CLEARED BY WRITING (VIA DATO) THE CORRESPONDING PAR
2260	013406	012706	001000		TESTS2: SCOPE		;CHECK EACH PDR
2261	013412	004767	002766		MOV	@KSTACK, SP	;INITIALIZE KERNEL STACK POINTER
2262	013416	000052			JSR	PC, ORDER	;CHECK TEST SEQUENCE + INIT SR0
2263	013420	104006			S2		;TEST NUMBER
2264	013422	004767	001766		HLT		;TEST EXECUTED OUT OF SEQUENCE
2265	013426	004767	003036		JSR	X7, RWALL	
2266	013432	012777	007600	165432	JSR	PC, KERN7	;MAP KERNEL PAR/PDR 7 TO EXT BANK
2267	013440	012737	140000	177776	MOV	@7600, @UPAR7	;MAP USER 7 TO EXTERNAL BANK
2268	013446	012706	000400		MOV	@140000, @#PS	;SET MODE TO USER
2269	013452	012700	001144		MOV	@USTACK, R6	;SETUP USER STACK
2270	013456	012001			MOV	@STATAB, R0	;SET UP KT REG TABLE POINTER
2271					LOP27: MOV	(R0)+, R1	;R1 CONTAINS ADDRESS OF
2272	013460	012702	017776		MOV	@17776, R2	;ADDRESSES OF CUR ENT PDR
2273							;R2 CONTAINS VIRTUAL ADDRESS TO
2274	013464	012037	177776		MOV	(R0)+, @#PS	;REFERENCE DESIRED PAGE
2275	013470	005277	165330		LOP27A: INC	@SR0	;SETUP STATUS FOR CURRENT MODE
2276	013474	011212			MOV	(R2), (R2)	;TURN ON MEMORY MANAG.
2277	013476	005077	165322		CLR	@SR0	;WRITE
2278	013502	004767	000016		JSR	X7, CKWBIT	;TURN OFF MEMORY MANAG.
2279	013506	062702	020000		ADD	@20000, R2	;TEST W BIT
2280	013512	103366			BCC	LOP27A	;CHANGE VA TO REFERENCE NEXT PAGE
2281							;LOOP UNTIL ALL PDR'S HAVE BEEN
2282	013514	020027	001152		CHP	R0, @STAEND	;CHECKED IN THE CURRENT MODE
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

2344	013776	032777	000100	165070	BIT	#100,2KPDRO	;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	#0,2KPARO	;DATOB TO THE PAR
2348	014016	032777	000100	165050	BIT	#100,2KPDRO	;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	2SR0	;TURN ON MEMORY MANAG.
2353	014034	013737	017776	017776	MOV	2#17776,2#17776	;WRITE INTO PAGE 0 AGAIN
2354	014042	005077	164756		CLR	2SR0	;TURN OFF MEMORY MANAG.
2355	014046	032777	000100	165020	BIT	#100,2KPDRO	;CHECK W BIT
2356	014054	001001			BNE	+.4	
2357	014056	104006			HLT		;W BIT NOT SET AFTER WRITING PAGE
2358	014060	016701	165030		MOV	KPARO,R1	;SETUP R1 TO REFERENCE HIGH BYTE
2359	014064	005201			INC	R1	;OF KPARO
2360	014066	112711	000000		MOVB	#0,2R1	;DATOB TO HIGH BYTE OF KPARO
2361	014072	032777	000100	164774	BIT	#100,2KPDRO	;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							
2367							
2368	014104	104400					
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 NUM JER
2372	014120	104006			HLT		;TEST EXECUTED OUT OF SEQUENCE
2373	014122	012767	000020	001526	MOV	#20,ICOUNT	
2374	014130	004767	001260		JSR	X7,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	#7600,2UPAR7	;MAP USER 7 TO THE EXTERNAL BANK
2377	014146	012737	140000	177776	MOV	#140000,2#PS	;SET MODE TO USER
2378	014154	012706	004000		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)+,2#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	2SR0	
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	2SR0	;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	#AORTAB,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	#100,2(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

;SHOW THAT THE W BIT IS NOT CLEARED BY INIT
;INITIALLY SET ALL THE W BITS, THEN DO A RESET AND CHECK THE W BITS

TESTS5: SCOPE

00	014246	005721			TST	(R1)+	: MOVE POINTER
01	014250	077207			SQB	R2, LOP32E	: CHECK ALL PDR'S IN THIS SET
02	014252	062701	000020		ADD	#20, R1	: CHANGE R1 TO REFERENCE NEXT
03							: SET OF PDR ADDRESSES
04	014256	020127	001132		CMP	R1, #ADREND	: CHECK FOR DONE
05	014262	002762			BLT	LOP32D	: IF NOT, CHECK NEXT SET OF PDR'S
06	014264	005037	177776		CLR	#PS	: SET MODE TO KERNEL
07	014270	005277	164530		INC	#SRO	: TURN MEMORY MANAG. ON
08	014274	000005			RESET		: INIT WITH MEMORY MANAG. ON
09	014276	000005			RESET		: INIT WITH MEMORY MANAG. OFF
10	014300	012701	001034		MOV	#ADRTAB, R1	: R1 REFERENCES ADDRESS OF PDR
11	014304	012702	000010		MOV	#10, R2	: R2 KEEPS TRACK OF WHEN TO CHANGE
12							: REGISTER SETS
13	014310	032771	000100	000000	LOP32G: BIT	#100, 2(R1)	: CHECK W BIT
14	014316	001001			BNE	.+4	
15	014320	104006			HLT		: INIT CLEARED W BIT IN PDR WHOSE
16							: ADDRESS IS POINTED TO BY R1
17	014322	005721			TST	(R1)+	: MOVE POINTER
18	014324	077207			SQB	R2, LOP32G	: CHECK ALL PDR'S IN THIS SET
19	014326	062701	000020		ADD	#20, R1	: CHANGE R1 TO REFERENCE NEXT SET
20							: OF PDR ADDRESSES
21	014332	020127	001132		CMP	R1, #ADREND	: CHECK FOR DONE
22	014336	002762			BLT	LOP32F	: IF NOT, CHECK NEXT SET OF PDR'S
23	014340	005077	164460		CLR	#SRO	: REINITIALIZE SRO
24							
25							
26							
27							
28							
29							
30							
31							
32							
33							
34							
35							
36							
37							
38							
39							
40							
41							
42							
43							
44							
45							
46							
47							
48							
49							
50							
51							
52							
53							
54							
55							

: SHOW THAT A DATO TO A PDR WILL CLEAR THE W BIT
 : EVEN WHEN THE INSTRUCTION ALSO CAUSES A TRAP REFERENCE TO
 : THE CORRESPONDING PAGE
 : MAP KERNEL PAGE 1 RRW AND MAKE A WRITE ACCESS TO PAGE 1
 : TO SET THE W BIT
 : THEN LOAD THE PDR, MAKING A TRAP REFERENCE TO PAGE 1 IN THE SOURCE
 : FETCH OF THE SAME INSTRUCTION-THE W BIT SHOULD BE CLEARED DUE
 : TO THE DATO TO THE PDR
 TEST56: SCOPE
 MOV #KSTACK, SP : INITIALIZE KERNEL STACK POINTER
 JSR PC, ORDER : CHECK TEST SEQUENCE + INIT SRO
 S6 : TEST NUMBER
 HLT : TEST EXECUTED OUT OF SEQUENCE
 MOV #2000, ICOUNT
 JSR X7, RWALL : INITIALIZE ALL PAGES RW, BANK 0
 JSR PC, KERN7 : MAP KERNEL PAR/PDR 7 TO EXT BANK
 MOV #77406, 2KPDR1 : MAKE KERNEL PAGE 1 RRW
 MOV #1, 2SRO : TURN ON MEMORY MANAG.
 MOV #20000, 2#20000 : READ AND WRITE PAGE 1
 CMP #77506, 2KPDR1 : CHECK PDR OF PAGE 1
 BEQ .+4
 : KERNEL PAGE 1 PDR
 : INCORRECT - W BIT SHOULD
 : BE SET DUE TO PREVIOUS MOVE INSTRUCTION
 : LOAD TEMP WITH VALUE TO BE MOVED TO KPDR1
 : PAGE 1 REFERENCE SHOULD SET
 : BUT DATO TO THE PDR CLEARS W BIT
 : CHECK PAGE 1 PDR
 MOV #77506, TEMP
 MOV TEMP+20000, 2KPDR1
 CMP #77406, 2KPDR1
 BEQ .+4
 HLT : PDR INCORRECT - W BIT
 CLR 2SRO : SHOULD HAVE BEEN CLEARED

K04

2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511

```

014466 104400
014470 012706 001000
014474 004767 001704
014500 000057
014502 104006
014504 012767 000010 001144
014512 004767 000676
014516 004767 001746
014522 012777 077406 164346
014530 012777 000001 164266
014536 013737 020000 020000
014544 022777 077506 164324
014552 001401
014554 104006
014556 012701 020000
014562 012702 000100
014566 005721
014570 077202
014572 022777 077506 164276
014600 001401
014602 104006
014604 005077 164214
014610 104400
014612 012706 001000
014616 004767 001562
014622 000060
014624 104006
014626 012767 002000 001022
014634 004767 000554
014640 004767 001624
014644 012777 014712 164312
014652 005077 164310
014656 012777 000001 164140
014664 012737 040000 177776
014672 000240
014674 005077 163076
014700 042777 000001 164116
014706 104006
014710 000415
014712 042777 000001 164104
014720 022777 100040 164076
014726 001401
014730 104006
014732 022777 014664 164072
014740 001401
014742 104006
    
```

```

;CHECK TO SEE THAT MULTIPLE ACCESSES TO A PAGE AFTER SETTING THE
;W BIT DON'T CLEAR THE W BIT
TEST57: SCOPE
MOV      #KSTACK, SP      ;INITIALIZE KERNEL STACK POINTER
JSR      PC, ORDER        ;CHECK TEST SEQUENCE + INIT SR0
S7       ;TEST NUMBER
HLT      ;TEST EXECUTED OUT OF SEQUENCE

MOV      #10, ICOUNT
JSR      %7, RWALL        ;INITIALIZE ALL PAGES 4K, RW, BANK 0
JSR      PC, KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
MOV      #77406, #KPDRI   ;MAP KERNEL PAGE 1 RW
MOV      #1, #SR0         ;TURN ON SEGMENTATION
MOV      #20000, #20000   ;READ AND WRITE PAGE 1
CMP      #77506, #KPDRI  ;CHECK THE PDR
BEQ      .+4
HLT      ;KERNEL PDR1 INCORRECT
          ;W BIT SHOULD BE SET

MOV      #20000, R1
MOV      #100, R2
L40:    TST      (R1)+      ;READ PAGE 1 REPEATEDLY
SOB     R2, L40
CMP      #77506, #KPDRI  ;CHECK W BIT AGAIN
BEQ      .+4
HLT      ;KERNEL PDR 1
          ;INCORRECT AFTER REPEATEDLY READING PAGE 1
          ;TURN OFF SEGMENTATION

;SHOW THAT IF MEMORY MANAG. IS ON, SETTING THE CURRENT MODE TO 10 WILL
;CAUSE A MEMORY MANAGEMENT ABORT. NON RESIDENT SHOULD BE SET, AND ALSO PL SHOULD
;BE SET
TEST60: SCOPE
MOV      #KSTACK, SP      ;INITIALIZE KERNEL STACK POINTER
JSR      PC, ORDER        ;CHECK TEST SEQUENCE + INIT SR0
B0       ;TEST NUMBER
HLT      ;TEST EXECUTED OUT OF SEQUENCE

MOV      #2000, ICOUNT    ;RESTORE ITERATION COUNT
JSR      %7, RWALL
JSR      PC, KERN7        ;MAP KERNEL PAR/PDR 7 TO EXT BANK
MOV      #RET42, #KTVEC   ;SETUP MEMORY MANAGEMENT ABORT RETURN
CLR      #KTSTA
MOV      #1, #SR0         ;TURN ON MEMORY MANAG.
MOV      #40000, #PS      ;SET MODE TO 01-FETCH OF NEXT
NOP      ;INSTRUCTION SHOULD ABORT
CLR      #PS             ;RESTORE MODE TO KERNEL
BIC      #1, #SR0        ;TURN OFF MEMORY MANAG.
HLT      ;NO ABORT WHEN MODE WAS SET
BR       CONT42          ;TO 01 (ILLEGAL)
BIC      #1, #SR0        ;TURN OFF MEMORY MANAG. AFTER ABORT
CMP      #100040, #SR0   ;CK SR0
BEQ      .+4
HLT      ;SR0 INCORRECT AFTER MODE 01 ABORT
          ;NR AND MODE 01 SHOULD BE SET

CMP      #A0042, #SR2    ;CHECK SR2
BEQ      .+4
HLT      ;SR2 INCORRECT - SHOULD CONTAIN
    
```

2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567

```

014744 005077 164054
014750 016777 164212 164206
104400
014756 104400
014760 012706 001000
014764 004767 001414
014770 000061
014772 104006
014774 004767 000414
015000 004767 001464
015004 012777 007723 164102
015012 012777 007776 164100
015020 005000
015022 012701 000042
015026 012702 040042
015032 052777 000400 163764 3S:
015040 012711 005252
015044 005077 163754
015050 027727 164042 005252
015056 001401
015060 104006
164030
015062 005077 164030
015066 052777 000400 163730 1S:
015074 012712 005252
015100 005077 163720
015104 027727 163746 005252
015112 001401
015114 104006
163734
015116 005077 163734 2S:
015122 022700 010000
015126 001415
015130 062700 000100
015134 062701 010100
015140 062702 000100
015144 162777 000001 163742
015152 162777 000001 163740
015160 000724
    
```

```

CONT42: CLR @SRO
MOV @TSTA,@KTVEC
    
```

```

*THIS TEST USES KPAR'S 0
*AND 2 TO REFERENCE KPAR1 AND UPAR1 RESPECTIVELY. A COUNT PATTERN IS
*RUN THROUGH THE VIRTUAL ADDRESS STARTING AT BIT6 AND THE RECIPROCAL
*COUNT PATTERN IS SIMULTANEOUSLY RUN THROUGH THE PAR'S. AFTER A
*RELOCATED REFERENCE IS MADE THE KT-11 IS TURNED OFF AND THE DATA IS
*CHECKED TO ENSURE THAT, WHATEVER THE CONDITION OF THE BITS IS IN THE
*VIRTUAL ADDRESS, THE DECODING FOR USER AND KERNAL PAR'S IS DONE BY
*THE PHYSICAL ADDRESS.
    
```

TEST61: SCOPE

```

MOV @KSTACK,SP
JSR PC,ORDER
61
HLT
JSR @X7,RWALL
JSR PC,KERN7
MOV @7723,@KPAR0
MOV @7776,@KPAR2
CLR R0
MOV @42,R1
MOV @40042,R2
BIS @400,@SRO
MOV @5252,(R1)
CLR @SRO
CMP @KPAR1,@5252
BEQ 1S
HLT
1S: CLR @KPAR1
BIS @400,@SRO
MOV @5252,(R2)
CLR @SRO
CMP @UPAR1,@5252
BEQ 2S
HLT
2S: CLR @UPAR1
CMP @10000,R0
BEQ EOP
ADD @100,R0
ADD @100,R1
ADD @100,R2
SUB @1,@KPAR0
SUB @1,@KPAR2
BR 3S
    
```

```

;ADDRESS OF THE INSTRUCTION
;IMMEDIATELY AFTER THE ONE SETTING
;THE MODE TO 01
;REINITIALIZE SRO
;RESTORE TRAP CATCHER

;INITIALIZE KERNEL STACK POINTER
;CHECK TEST SEQUENCE + INIT SRO
;TEST NUMBER
;TEST EXECUTED OUT OF SEQUENCE
;SETUP ALL PAR'S FOR 4K R/W
;SET UP KERNAL 7 REGISTERS
;LOAD KPAR0 WITH ADDR OF KPAR1
;LOAD KPAR2 WITH ADDR OF UPAR1
;CLEAR COUNTER REGISTER
;LOAD OFFSET & BIT TO SELECT KPAR0
;LOAD OFFSET & BIT TO SELECT KPAR2
;TURN ON MAINTENANCE MODE
;LOAD PATTERN IN KERNAL PAR1
;TURN OFF MAINTENANCE MODE
;DID DATA GET STORED IN KPAR1?
;BRANCH IF DATA STORED CORRECTLY
;A HALT HERE INDICATES THAT THE
;RELOCATION TO KPAR1 WAS NOT
;SUCCESSFUL R1 HAS VIRTUAL ADDR AND
;KPAR0 HAS THE BASE.
;CLEAR KPAR1 FOR NEXT TEST
;TURN ON MAINTENANCE MODE
;LOAD PATTERN IN USER PAR1
;TURN OFF MAINTENANCE MODE
;DID DATA GET STORED IN UPAR1?
;BRANCH IF DATA STORED CORRECTLY
;A HALT HERE INDICATES THAT THE
;RELOCATION TO UPAR1 DID NOT WORK
;R2 HAS THE VIRTUAL ADDR AND KPAR2
;HAS THE BASE
;CLEAR UPAR1 FOR NEXT TEST
;CHECK TO SEE IF TEST IS DONE
;BRANCH IF TEST IS OVER
;ADD BIT6 TO COUNTER
;ADD BIT6 TO KPAR1'S VIRTUAL ADDR
;ADD BIT6 TO UPAR1'S VIRTUAL ADDR
;SUBTRACT BIT1 FROM KPAR1'S BASE
;SUBTRACT BIT1 FROM UPAR1'S BASE
;CONTINUE TEST
    
```

```

2568 015162 104400 EOP: SCOPE
2569 015164 032777 010000 164006 BIT #BIT12,2SR ;INHIBIT BELL?
2570 015172 001003 BNE 1$ ;BRANCH IF BELL IS INHIBITED
2571 015174 012700 015360 MOV #BELL,RO ;PUT ADDRESS OF BELL CHARS IN RO
2572 015200 000402 BR 2$ ;BRANCH TO OUTPUT CODE
2573 015202 012700 015364 1$: MOV #ASTER,RO ;PUT ADDRESS OF ASTERICK CHARS IN RO
2574 015206 112001 2$: MOVB (RO)+,R1 ;CHECK FOR TERMINATOR CODE
2575 015210 001405 BEQ LOGICT ;BRANCH IF BYTE IS ZERO
2576 015212 010177 163602 3$: MOV R1,2TDBR ;OUTPUT CHARACTER TO BUFFER
2577 015216 105777 163574 TSTB 2TCSR ;SEE IF STATUS REG GETS SET
2578 015222 100373 BPL 3$ ;BRANCH UNTIL IT DOES
2579 015224 013701 000042 LOGICT: MOV 2#42,R1 ;MONITOR HOOK
2580 015230 001405 BEQ END
2581 015232 000005 RESET
2582 015234 004711 LOGIC: JSR PC,2R1
2583 015236 000240 NOP
2584 015240 000240 NOP
2585 015242 000240 NOP
2586 015244 000167 163734 END: JMP START

;MESSAGE AREA
2589 MTIT: .ASCII <15><12>'11/34 MEMORY MANAGEMENT LOGIC TEST MAINDEC-11-DFKTA-A'<15><12>'
2590 015250 005015 030461 031457
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= 2'
2601 015350 100
2602 015351 040 050040 036523 MPS: .ASCII ' PS= 2'
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 2SR0
2611 015374 005000 CLR RO
2612 015376 012701 000040 MOV #32,R1 ;COUNT OF REGISTERS TO BE CLEARED
2613 015402 005070 001034 CLRLP: CLR 2ADRTAB(RO) ;CLEAR REGISTERS THRU ADDRESS TABLE
2614 015406 005720 TST (RO)+ ;MOVE POINTER
2615 015410 077104 SOB R1,CLRLP ;LOOP TILL DONE
2616 015412 000207 RTS %?
2617
2618 ;SUBROUTINE TO MAKE ALL PAGES RW, BANK 0, 4K, UP
2619 015414 005077 163404 RWALL: CLR 2SR0
2620 015420 012701 001034 MOV #ADRTAB,R1 ;R1 POINTS TO ADDRESS TABLE
2621 015424 012700 000010 RWL1: MOV #10,RO ;RO IS COUNTER
2622 015430 005071 000020 RWL2: CLR 220(R1) ;CLEAR PAR
2623 015434 012731 077406 MOV #77406,2(R1)+ ;SET PDR RW, 4K

```

2624	015440	077005			SOB	RO,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	X7	
2629							
2630							
2631							
2632							
2633							
2634							
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
2646	015536	000177	000000		JMP	#RETRNX	; KEEP COUNT AT ZERO
2647	015542	000000			RETRNX:	0	; JUMP TO TEST
2648							
2649	015544	032777	040000	163426	SCOPE AND/OR ITERATION LOOP FOR EACH TEST 4000 TIMES		
2650	015552	001015			SCOPEC:	BIT	#BIT14,#SR
2651	015554	032777	004000	163416	BNE	SCOPEB	; TEST SR FOR SCOPE
2652	015562	001020			BIT	#BIT11,#SR	; YES SCOPE
2653	015564	026767	000070	000064	BNE	SCOPEG	; NO-TEST FOR ITERATION
2654	015572	100014			CMP	SCOPEF,ICOUNT	; INHIBIT ITERATION
2655	015574	005267	000060		BPL	SCOPEG	; COMPARE CURRENT COUNT TO MAX NUMBER
2656	015600	012737	000340	177776	INC	SCOPEF	; EXIT-DONE
2657	015606	022626			MOV	#340,#PS	; INCREMENT COUNT
2658	015610	005037	177776		SCOPEB:	CMP	(6)+,(6)+
2659	015614	005077	163204		CLR	#PS	; PREVENT TRAPPING WHILE MOVING STACK
2660	015620	000177	000036		CLR	#SR0	; REPOSITION STACK
2661	015624	005067	000030		JMP	#RETURN	
2662	015630	005267	163340		SCOPEG:	CLR	SCOPEF
2663					INC	TESTCT	; REPEAT TEST
2664	015634	011667	000022				; CLEAR COUNT
2665	015640	022626			MOV	#6,RETURN	; STEP TEST COUNTER TO ALLOW CHECKING
2666	015642	005037	177776		CMP	(6)+,(6)+	; ORDER OF EXECUTION.
2667	015646	005077	163152		CLR	#PS	; SAVE SCOPE RETURN POINTER
2668	015652	000177	000004		CLR	#SR0	; RETURN INLINE-NEXT TEST
2669	015656	004000			JMP	#RETURN	
2670	015660	000000			ICOUNT:	4000	; ITERATION COUNT
2671	015662	000000			SCOPEF:	0	; COUNT LOCATION FOR ITERATION LOOP
2672					RETURN:	0	; ADDRESS OF LAST TEST
2673							
2674							
2675	015664	012767	000340	162104	PRINT:	MOV	#340,PS
2676	015672	032777	020000	163300	BIT	#BIT13,#SR	; SET PRIORITY TO 7
2677	015700	001401			BEQ	.+4	; TEST FOR INHIBIT PRINT OUT
2678	015702	000430			BR	CK	; BRANCH TO PRINT
2679	015704	012667	000066		MOV	(6)+,SAVPC	; INHIBIT CHECK FOR HALT

DFKTAA.P11

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		TST	SA	;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							
2699	016002	010067	000052				
2700	016006	011600			TYPE: MOV	X0,SAVR0	
2701	016010	062716	000002		MOV	(6),X0	;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
2702	016014	011000			ADD	#2,X0	;SET UP EXIT
2703	016016	112067	000034		MOV	X0,X0	
2704	016022	122767	000100	000026	TYPB: MOVB	(0),TYPDAT	;GET CHARACTER
2705	016030	001003			CMPB	#100,TYPDAT	;CHECK FOR "3" CHARACTER
2706	016032	016700	000022		BNE	TYPB	;BRANCH IF NOT "3"
2707	016036	000207			MOV	SAVR0,X0	;RESTORE R0
2708	016040	116777	000012	162752	RTS	PC	;TERMINATOR CHAR. EXIT
2709	016046	105777	162744		TYPB: MOVB	TYPDAT,@TDBR	;OUTPUT CHAR TO PRINTER
2710	016052	100375			TSTB	@TCSR	;WAIT FOR TTY READY
2711	016054	000760			BPL	.-4	
2712	016056	000000			BR	TYPB	
2713	016060	000000			TYPDAT:	0	
2714					SAVR0:	0	
2715							
2716							
2717							
2718	016062	012767	000001	000232			
2719	016070	005767	000232		PRSHRT: MOV	#1,PRSHRT	;SET FLAG TO INDICATE SHORT PRINTOUT
2720	016074	001011			TST	PTEMP1	;CHECK FOR ZERO
2721	016076	012777	000260	162714	BNE	PROCT+4	;BRANCH IF NOT ZERO
2722	016104	105777	162706		MOV	#260,@TDBR	;OUTPUT A SINGLE ZERO
2723	016110	100375			TSTB	@TCSR	;WAIT FOR TTY READY
2724	016112	000207			BPL	.-4	
2725	016114	005067	000202		RTS	X7	;RETURN
2726	016120	005067	000206		PROCT: CLR	PRSHRT	;CLEAR FLAG TO INDICATE FULL PRINTOUT
2727	016124	005067	000174		CLR	PTEMP3	;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
2728	016130	012767	000260	000172	CLR	PRFLG	;INITIALIZE CARRY FLAG FOR ROTATES
2729	016136	005767	000164		MOV	#260,PTEMP2	;SETUP R3
2730	016142	100002			TST	PTEMP1	;CHECK BIT 15 OF NUMBER
2731	016144	005267	000160		BPL	.+6	;BRANCH IF ZERO
2732	016150	006167	000152		INC	PTEMP2	;INCREMENT R3 IF ONE
2733	016154	006167	000146		ROL	PTEMP1	;ROTATE LEFT MOST OCTAL TO RIGHT END
2734	016160	005567	000140		ROL	PTEMP1	
2735	016164	005767	000132		ADC	PRFLG	;STORE CARRY
2736	016170	001404			P.CK: TST	PRSHRT	;CHECK FOR SHORT PRINTOUT
2737	016172	026727	000132	000260	BEQ	P.WAIT	;BRANCH IF NOT SET
2738	016200	001410			CMP	PTEMP2,#260	;CHECK FOR ZERO IF SET
2739	016202	016777	000122	162610	BEQ	P.CONT	;IF SET, GO TO NEXT CHARACTER
					P.WAIT: MOV	PTEMP2,@TDBR	;OUTPUT NEXT CHARACTER

2740	016210	105777	162602		TSTB	BTCSR		;WAIT FOR TTY READY
2741	016214	100375			BPL	-4		
2742	016216	005067	000100		CLR	PRFLG		;PRINT REST OF NUMBER AFTER A NON-ZERO DIGIT
2743	016232	005267	000104		INC	PTEMP3		;COUNT
2744	016236	026727	000100	000006	CMR	PTEMP3, #6		;CHECK FOR DONE
2745	016234	001001			BNE	P, CNT1		;BRANCH IF NOT DONE
2746	016236	000207			RTS	x7		
2747	016240	000241			CLC			;CLEAR CARRY
2748	016242	000767	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	000167	000044		ROL	PTEMP1		;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTE
2753	016258	000167	000040		ROL	PTEMP1		
2754	016256	000167	000034		ROL	PTEMP1		
2755	016272	000567	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			PRFLG:	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								
2767								
2768	016334	011667	000032					
2769	016340	162767	000002	000024	EMTSRV:	MOV	2SP, EPC	;GET CALL
2770	016346	017767	000020	000016		SUB	#2, EPC	
2771	016354	105067	000013			MOV	2EPC, EPC	
2772	016360	062767	016374	000004		CLRF	EPC+1	;SAVE OFFSET ONLY
2773	016366	017707	000000			ADD	2ENTAB, EPC	;POINT TO TABLE OF ADDRESSES
2774	016372	000000				MOV	2EPC, PC	;JUMP TO DESIRED ROUTINE
2775		104000			EPC:	0		
2776		104002				PATCH1=ENT+0		
2777		104004				PATCH2=ENT+2		
2778	016374	104000				PATCH3=ENT+4		
2779	016376	104002			ENTAB:	PATCH1		
2780	016400	104004				PATCH2		
2781	016402	015664				PATCH3		
2782						PRINT		
2783								
2784	016404	005037	177776					
2785	016410	011667	000052					
2786	016414	017767	000046	000044				
2787	016422	032777	002000	162550				
2788	016430	001404						
2789	016432	016700	000030					
2790	016436	000005						
2791	016440	000000						
2792	016442	026767	162526	000016	OPDERB:	CMR	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	

E05

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 56
DFKTA.A.P11

2796	016460	062716	000004
2797	016464	000007	
2798	016466	000000	
2799			
2800	016470	012777	007600 162434
2801	016476	012777	077406 162406
2802	016504	000207	
2803		017712	
2804			
2805	017712	125252	
2806		000001	

```

ORDERA: AOO      M4, (SP)
         RTS      PC
TEMPN:  0
:MAP KERNEL PAR/PDR 7 TO EXTERNAL BANK
KERN7:  MOV      @7600, @KPAR7
         MOV      @77406, @KPDR7
         RTS      PC
         .=17712

DESTAD: 125252
        .END

```

;UPDATE FOR GOOD RETURN

CNT22E	004546	1060	1069#																	
CNT31A	005570	1239	1241#																	
CNT31B	005560	1254	1260#																	
CONT11	012202	2015	2024#																	
CONT23	012564	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#																
DONE5	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#													
ENTAB	016374	2772	2778#																	
ENTSRV	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#												
ERR29	004264	993	1000#																	
ERR38	004350	1012	1019#																	
ERR2C	004436	1032	1039#																	
ERR2D	004532	1053	1061#																	
ERR2E	004630	1076	1084#																	
EXIT10	007360	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						

H05

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 60
 DFKTAA.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

		1515	1520	1572	1579	1592	1602	1607	1614	1619	1622	1625	1634	1639
		1642	1646	1660	1670	1675	1677	1687	1690	1693	1702	1707	1710	1714
		1728	1738	1743	1750	1755	1758	1761	1770	1775	1778	1782	1793	1797
		1801	1809	1828	1834	1846	1850	1865	1875	1892	1887	1897	1917	1936
		1945	1951	1957	1963	1973	1982	2000	2019	2022	2033	2043	2050	2053
		2063	2073	2076	2078	2109	2113	2121	2125	2145	2165	2170	2177	2181
		2198	2220	2236	2248	2254	2253	2287	2293	2304	2312	2316	2323	2329
		2337	2346	2350	2357	2363	2372	2396	2415	2437	2446	2454	2463	2472
		2480	2491	2502	2507	2511	2530	2543	2553					
ICOUNT	015656	423*	446*	483*	574*	582*	637*	708*	756*	777*	861*	917*	1097*	1137*
		1223*	1872*	1867*	2199*	2249*	2373*	2438*	2464*	2492*	2653	2669*		
INT25	005334	1178	1186*											
INT35	007010	1465	1468*											
INT36	007576	1597	1604*											
INT36A	007652	1508	1616*											
INT36B	007752	1627	1636*											
INT37	010132	1665	1672*											
INT37A	010206	1676	1684*											
INT37B	010306	1695	1704*											
INT40	010466	1733	1740*											
INT40A	010542	1744	1752*											
INT40B	010642	1763	1772*											
IOT35	006606	1416	1419*											
KERN7	016470	810	921	984	1101	1141	1177	1225	1287	1318	1366	1594	1662	1730
		1813	1830	1877	1899	1938	2004	2036	2065	2090	2147	2201	2265	2306
		2339	2375	2440	2466	2494	2532	2800*						
KPDR0	001114	374*	779*	805*	919*	1099*	1139*	2347*	2358	2533*	2563*			
KPDR1	001116	375*	979*	1007*	1026*	1046*	1069*	2541	2547*					
KPDR2	001120	376*	2534*	2564*										
KPDR3	001122	377*												
KPDR4	001124	378*												
KPDR5	001126	379*												
KPDR6	001130	380*												
KPDR7	001132	381*	2800*											
KPDR8	001074	365*	389	754*	780*	807*	920*	982*	1100*	1140*	1176*	1203	1313*	1595*
		1663*	1731*	1811*	2002*	2252	2310	2313*	2314	2321	2324	2327	2344	2348
		2355	2361											
KPDR1	001076	366*	983*	1204	1286*	1596*	1664*	1732*	1814*	1939*	2098*	2155*	2208*	2441*
		2444	2450*	2452	2467*	2470	2478							
KPDR2	001100	367*	2035*	2051										
KPDR3	001102	368*												
KPDR4	001104	369*												
KPDR5	001106	370*												
KPDR6	001110	371*												
KPDR7	001112	372*	2071	2801*										
KRET34	006172	1315	1324*	1332										
KSTACK	001000	331*	411	435	451	479	511	539	578	606	633	669	704	716
		747	773	800	849	913	977	1093	1133	1171	1219	1281	1304	1350
		1359	1372	1589	1657	1725	1790	1806	1843	1872	1894	1933	1970	1997
		2030	2060	2085	2142	2195	2245	2260	2301	2334	2369	2434	2460	2488
		2527	2636											
KTSTA	001166	400*	907	908*	1179*	1211*	1212	1289*	1298	1598*	1649*	1650	1666*	1717*
		1718	1734*	1765*	1786	1816*	1836	1852*	1868	1901*	1926	1927*	1965	2006*
		2025	2038*	2054	2067*	2078	2096*	2133	2134*	2153*	2188	2189*	2203*	2240
		2496*	2516											
KTVEC	001164	399*	907*	1178*	1212*	1288*	1298*	1597*	1608*	1627*	1650*	1665*	1676*	1695*

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#			

E06

DFKTA.A MACY11 27(732) 09-SEP-76 17:12 PAGE 72
 DFKTAA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

RTS	2295	2616	2628	2707	2724	2746	2795	2797	2802						
RTT	2009														
SEC	2751														
S08	501	502	533	564	601	637	628	629	645	646	657	658	680	681	692
	693	1236	1691	1935	2401	2418	2477	2615	2624						
SUB	2183	2184	2563	2564	2769										
SUB	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	2099	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												
.ASCII2	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	1290	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													
.MLIST	1	307	408	434	450	478	510	538	577	605	632	668	703	715	746
	772	799	848	912	976	1092	1132	1170	1218	1290	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

#DFKTA.A DFKTAA.S08/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DFKTA.A.P11
 RUN-TIME: 10 20 4 SECONDS
 RUN-TIME RATIO: 77/36=2.1
 CORE USED: 10K (19 PAGES)

