

KT11-D

KT11-D STATES
MD-11-DBKTD-C

EP-DBKTD-C-DL-C
COPYRIGHT © 1977
FICHE 1 OF 1

APR 1977
digital
MADE IN USA

This microfiche card contains a grid of frames. The frames are arranged in approximately 10 rows and 2 columns. Each frame contains a table of data, likely representing a state's statistics for the year 1977. The data is organized into columns and rows, with some frames containing headers and footnotes. The text is small and difficult to read, but the overall structure is consistent across the frames.

801

EOF1DBKTCBSEQ

00010000

770323

PDP10 411

HDR1DBKTCBSEQ

00010000

770323

CO1

DBKTD-C KT11-D PROCESSORS STATES TEST MACY11 27(1006) 02-FEB-77 10:09 PAGE 2
DBKTD.C.P11 02-FEB-77 09:11

.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DBKTD-C-D
PRODUCT NAME: KT11-D PROCESSORS STATES TEST
DATE RELEASED: MARCH, 1977
MAINTAINER: DIAGNOSTIC GROUP

COPYRIGHT 1972, 1977 BY DIGITAL EQUIPMENT CORPORATION
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
DOCUMENT.
THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A
LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH
THE TERMS OF SUCH LICENSE.
DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT
THAT IS NOT SUPPLIED BY DIGITAL.

1.0 ABSTRACT

THIS IS A TEST THAT UTILIZES THE KT11-D MEMORY MANAGEMENT OPTION AND TESTS THAT IN THE TWO PDP-11/40 STATES (KERNEL, USER) INSTRUCTIONS ARE EXECUTED PROPERLY. THIS TEST TESTS TRAPS FROM ONE STATE TO THE OTHER AND USES THE MFPI/MTPI INSTRUCTIONS.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11/40 WITH KT11-D (MEM. MGMT.) INSTALLED.

2.2 STORAGE

UTILIZES 4K OF MEMORY

3.0 LOADING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABSOLUTE LOADER. PROGRAM MAY ALSO BE LOADED VIA XXDP OR ACT11.

4.0 STARTING PROCEDURE

LOAD ADDRESS 200. PRESS START, THE PROGRAM WILL LOOP AND RING BELL AND PRINT AN '*' ON PASS COMPLETION.

5.0 OPERATION PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

NONE

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

SCOPE IS A MOV PC,R1 AND STORE THE PC+2 IN R1; THUS R1 MAY BE USED AS A REFERENCE TO DETERMINE THE LAST TEST SUCCESSFULLY COMPLETED.

5.2.2 HLT

HLT IS A HALT INSTRUCTION AND IS EXECUTED WHENEVER A HARDWARE MALFUNCTION IS DETECTED.

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 PASS COUNT (ICNT)

THE NUMBER OF PROGRAM PASSES COMPLETED IS CONTAINED IN ADDRESS ICNT (LOC. 1000). THIS ADDRESS MAY BE EXAMINED TO DETERMINE IN WHICH PASS THE ERROR OCCURED.

6.0 ERRORS

6.1 TEST ERROR WILL CAUSE A HALT

FALSE TRAP/INTERRUPT ERRORS - THE PROGRAM WILL HALT AT THE TRAP VECTOR ADDRESS +2. THE CONTENTS OF R6 CONTAINS THE ADDRESS WHERE THE PC OF THE INSTRUCTION THAT CAUSED THE TRAP IS STORED.

6.2 ERROR RECOVERY

TEST ERRORS - PRESS CONTINUE OR LOOP TEST (SEE 6.3)
TRAP ERRORS - DETERMINE WHERE ERROR OCCURED (SEE 6.1)

6.3 ERROR LOOPING

TO LOOP ON AN ERROR REPLACE THE HLT INSTRUCTION WITH A BRANCH BACK TO THE PREVIOUS SCOPE INSTRUCTION. NOTE THAT IF THE ERROR IS INTERMITTENT THE TEST WILL DROP THROUGH THE HLT AND PROCEED TO THE NEXT TEST. THEREFORE, TO LOOP THE TEST CONTINUOUSLY, REPLACE THE BEQ +4 INSTRUCTION PRECEEDING THE HLT WITH THE BRANCH BACK TO THE PREVIOUS SCOPE.

7.0 RESTRICTIONS

THIS PROGRAM MUST BE LOADED IN LOWER 4K.

7.1 STARTING RESTRICTION

ALL PROGRAMS MUST BE INITIALLY STARTED AT 200 AND MAY BE STARTED AT A SCOPE INSTRUCTION THEREAFTER.

7.2 OPERATIONAL RESTRICTIONS

NONE

8.1 EXECUTION TIME

ONE PASS TAKES APPROXIMATELY 10 SECONDS.

F01

DBKTD-C KT11-D PROCESSORS STATES TEST MACY11 27(1006) 02-FEB-77 10:09 PAGE 5
DBKTDC.P11 02-FEB-77 09:11

%

150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169

```

170
171 ; TEST DBKTD-C TESTS FEATURES OF THE TWO PROCESSOR STATES AND INCLUDES
172 ; TRAPS FROM ALL STATES TO ALL OTHER STATES, AND MFP/MTP INSTRUCTIONS IN ALL
173 ; STATES AND PREVIOUS STATES.
174 ; NOTE: ALL TESTS ARE ENTERED AND EXITED IN KERNEL MODE.
175
176 ; STARTING PROCEDURE
177 :   LOAD ADDRESS=200
178 :   PRESS START
179 :   KERNEL STACK POINTER IS AT 500
180 :   USER STACK POINTER IS AT 700
181 :   BELL WILL RING WHEN TEST IS COMPLETE
182
183 ; REGISTER ASSIGNMENTS
184 000000 R0=%0
185 000001 R1=%1
186 000002 R2=%2
187 000003 R3=%3
188 000004 R4=%4
189 000005 R5=%5
190 000007 PC=%7
191
192 ; STACK POINTERS
193 000006 KSP=%6 ; KERNEL STACK POINTER
194 000006 USP=%6 ; USER STACK POINTER
195 000000 HLT=HALT
196 010701 SCOPE=010701 ; MOVE PC TO R1
197 000003 TRT=3 ; TRACE TRAP
198 000140 PRTY3=140
199 000200 PRTY4=200
200 000340 PRTY7=340
201
202 ; VECTOR ADDRESSES
203 000004 ERRVEC=4 ; ADDRESS OF ERROR VECTOR
204 000010 RESVEC=10 ; ADDRESS OF RESERVED INST TRAP VECTOR
205 000030 EMTVEC=30 ; ADDRESS OF EMT VECTOR
206 000034 TRAPVEC=34 ; ADDRESS OF TRAP VECTOR
207 000020 IOTVEC=20 ; ADDRESS OF IOT VECTOR
208 000014 TBITVEC=14 ; ADDRESS OF 'T' BIT TRAP VECTOR
209 000014 TRTVEC=14 ; ADDRESS OF 'TRACE' TRAP
210 000064 TPVEC=64 ; ADDRESS OF TTY PRINTER INTERRUPT VECTOR
211
212 ; HARDWARE REGISTER ASSIGNMENTS
212 177776 PSW=177776 ; ADDRESS OF STATUS REGISTER
213 177774 SLR=177774 ; ADDRESS OF STACK LIMIT REGISTER
214 177560 TKS=177560 ; ADDRESS OF KEYBOARD CSR
215 177562 TKB=177562 ; ADDRESS OF KEYBOARD BUFFER
216 177564 TPS=177564 ; ADDRESS OF TELEPRINTER CSR
217 177566 TPB=177566 ; ADDRESS OF TELEPRINTER BUFFER
218 177570 SWR=177570 ; ADDRESS OF CONSOL SWITCH REGISTER
219
220 ; INITIAL STACK POINTER SETTINGS
220 000500 KPTR=500 ; KERNEL INITIAL STACK POINTER VALUE
221 000700 UPTR=700 ; USER INITIAL STACK POINTER VALUE
222 001000 YELPTR=1000 ; STACK POINTER VALUE FOR 'YELLOW' OVFLW
223 000736 REDPTR=736 ; STACK POINTER VALUE FOR 'RED' OVFLW
224
225 ; MISC. BIT ASSIGNMENTS
    
```


282	000104	000106	.+2
283	000106	000000	HALT
284	000110	000112	.+2
285	000112	000000	HALT
286	000114	000116	.+2
287	000116	000000	HALT
288	000120	000122	.+2
289	000122	000000	HALT
290	000124	000126	.+2
291	000126	000000	HALT
292	000130	000132	.+2
293	000132	000000	HALT
294	000134	000136	.+2
295	000136	000000	HALT
296	000140	000142	.+2
297	000142	000000	HALT
298	000144	000146	.+2
299	000146	000000	HALT
300	000150	000152	.+2
301	000152	000000	HALT
302	000154	000156	.+2
303	000156	000000	HALT
304	000160	000162	.+2
305	000162	000000	HALT
306	000164	000166	.+2
307	000166	000000	HALT
308	000170	000172	.+2
309	000172	000000	HALT
310	000174	000176	.+2
311	000176	000000	HALT
312	000200	000202	.+2
313	000202	000000	HALT
314	000204	000206	.+2
315	000206	000000	HALT
316	000210	000212	.+2
317	000212	000000	HALT
318	000214	000216	.+2
319	000216	000000	HALT
320	000220	000222	.+2
321	000222	000000	HALT
322	000224	000226	.+2
323	000226	000000	HALT
324	000230	000232	.+2
325	000232	000000	HALT
326	000234	000236	.+2
327	000236	000000	HALT
328	000240	000242	.+2
329	000242	000000	HALT
330	000244	000246	.+2
331	000246	000000	HALT
332	000250	000252	.+2
333	000252	000000	HALT
334	000254	000256	.+2
335	000256	000000	HALT
336	000260	000262	.+2
337	000262	000000	HALT

338	000264	000266		.+2		
339	000266	000000		HALT		
340	000270	000272		.+2		
341	000272	000000		HALT		
342	000274	000276		.+2		
343	000276	000000		HALT		
344	000300	000302		.+2		
345	000302	000000		HALT		
346	000304	000306		.+2		
347	000306	000000		HALT		
348	000310	000312		.+2		
349	000312	000000		HALT		
350	000314	000316		.+2		
351	000316	000000		HALT		
352	000320	000322		.+2		
353	000322	000000		HALT		
354	000324	000326		.+2		
355	000326	000000		HALT		
356	000330	000332		.+2		
357	000332	000000		HALT		
358	000334	000336		.+2		
359	000336	000000		HALT		
360	000340	000342		.+2		
361	000342	000000		HALT		
362	000344	000346		.+2		
363	000346	000000		HALT		
364	000350	000352		.+2		
365	000352	000000		HALT		
366	000354	000356		.+2		
367	000356	000000		HALT		
368	000360	000362		.+2		
369	000362	000000		HALT		
370	000364	000366		.+2		
371	000366	000000		HALT		
372	000370	000372		.+2		
373	000372	000000		HALT		
374	000374	000376		.+2		
375	000376	000000		HALT		
376		000046		.=46		
377	000046	006624		\$ENDAD		
378		000052		.=52		
379	000052	000000		000000		
380						
381		000200		.=200		
382	000200	000167	000612	JMP	START	;GO START
383						
384		001000		.=1000		
385						
386						
387	001000	000000				
388	001002	000000				
389		001012				
390	001012	000000				
391	001014	000000				

;TAGS

ICNT: 0
 TEMP: 0

;CONTAINS PASS COUNT

;TITLE FLAG

```

392
393 001016 012706 000500 START: MOV #KPTR,KSP
394 001022 005067 177752 CLR ICNT
395 001026 005767 177760 TST FTITLE ;HAS TITLE BEEN PRINTED YET?
396 001032 001050 BNE PWRUP ;YES, SKIP TITLE
397 001034 023737 000042 000046 CMP @#42,@#46 ;ARE WE IN ACT11 AUTOMATIC MODE?
398 001042 001444 BEQ PWRUP ;YES, SKIP TITLE
399 001044 012700 001076 MOV #TITLE,RO ;GET MESSAGE ADDRESS
400 001050 012767 000001 177734 MOV #1,FTITLE ;SET FLAG
401 001056 105767 176502 IS: TSTB TPS
402 001062 100375 BPL IS
403 001064 105710 TSTB (0) ;END OF MESSAGE?
404 001066 001432 BEQ PWRUP ;YES, GET OVER THE ASCII
405 001070 112067 176472 MOVB (0)+,TPB ;PRINT CHARACTER
406 001074 000770 BR IS ;GO DO THE NEXT ONE
407 001076 005015 042120 030520 TITLE: .ASCIZ <15><12>@PDP11/40 PROCESSOR STATES TEST, DBKTD-C<15><12>·177>
408 001104 027461 030064 050040
409 001112 047522 042503 051523
410 001120 051117 051440 040524
411 001126 042524 020123 042524
412 001134 052123 020054 041104
413 001142 052113 026504 006503
414 001150 077412 000
415 001154
416 .EVEN
417 001154 032737 00000C 177776 PWRUP: BIT #KM+PKM,@#PSW ;TEST THAT PROCESSOR POWERED UP OK FOR THE TEST
418 001162 001377 BNE . ;IS STATUS CORRECT
419 ;LOOP HERE IF NOT
420 001164 012706 000500 BEGIN: MOV #KPTR,KSP ;INITIALIZE THE STACK POINTER
421
422 ;CHECK THAT THE NOP INSTRUCTION IS A 'NOP' IN USER MODE.
423 001170 010701 †1: SCOPE
424 001172 012737 140000 177776 MOV #UM,@#PSW ;USER MODE,PRIORITY LEVEL 0
425 001200 000240 NOP
426 001202 013700 177776 MOV @#PSW,RO ;GET @#PSW
427 001206 005037 177776 CLR @#PSW ;KERNEL MODE!!!
428 001212 022700 140000 CMP #UM,RO ;TEST THAT NOP DID NOT ALTER @#PSW
429 001216 001401 BEQ .+4
430 001220 000000 HLT ;ERROR! NOP CHANGED STATUS WORD
431
432
433 ;TEST TRAP FROM USER MODE TO KERNEL MODE
434 001222 010701 †5: SCOPE
435 001224 012706 000500 MOV #KPTR,KSP
436 001230 012737 001266 000020 MOV #TSA,@#IOTVEC
437 001236 005067 176560 CLR IOTVEC+2
438 001242 012737 140340 177776 MOV #UM+PRTY7,@#PSW ;USER MODE!!!
439 001250 012706 000700 MOV #UPTR,USP
440 001254 000277 SCC
441 001256 000004 IOT
442 001260 005037 177776 TSAA: CLR @#PSW
443 001264 000000 HLT
444 001266 013700 177776 TSA: MOV @#PSW,RO
445 001272 005037 177776 CLR @#PSW
446 001276 022700 030000 CMP #KM+PUM,RO
447 001302 001401 BEQ .+4
    
```

448	001304	000000			HLT	
449	001306	022767	001260	177160	CMP	#T5AA, KPTR-4
450	001314	001401			BEQ	.+4
451	001316	000000			HLT	
452	001320	022767	140357	177150	CMP	#UM+PRTY7+17, KPTR-2
453	001326	001401			BEQ	.+4
454	001330	000000			HLT	
455	001332	022706	000474		CMP	#KPTR-4, KSP
456	001336	001401			BEQ	.+4
457	001340	000000			HLT	
458	001342	012737	140000	177776	MOV	#UM, @PSW
459	001350	010600			MOV	USP, R0
460	001352	005037	177776		CLR	@PSW
461	001356	022700	000700		CMP	#UPTR, R0
462	001362	001401			BEQ	.+4
463	001364	000000			HLT	
464	001366	012737	000022	000020	MOV	#IOTVEC+2, @IOTVEC
465						
466						: TEST TRAP FROM USER TO USER MODE (VIA TRACE TRAP)
467	001374	010701			T7:	SCOPE
468	001376	012767	001434	176410	MOV	#T7A, TRTVEC
469	001404	012767	140000	176404	MOV	#UM, TRTVEC+2 ;USER MODE ON TRAP
470	001412	012737	140000	177776	MOV	#UM, @PSW
471	001420	012706	000700		MOV	#UPTR, USP
472	001424	000003			TRT	
473	001426	005037	177776		T7AA:	CLR @PSW
474	001432	000000			HLT	
475	001434	013700	177776		T7A:	MOV @PSW, R0
476	001440	010602			MOV	USP, R2
477	001442	042737	140000	177776	BIC	#UM, @PSW
478	001450	022767	001426	177216	CMP	#T7AA, UPTR-4
479	001456	001401			BEQ	.+4
480	001460	000000			HLT	
481	001462	022700	170000		CMP	#UM+PUM, R0
482	001466	001401			BEQ	.+4
483	001470	000000			HLT	
484	001472	012767	000016	176314	MOV	#TRTVEC+2, TRTVEC
485	001500	005067	176312		CLR	TRTVEC+2
486						
487						: TEST THAT THE 'HALT' INSTRUCTION TRAPS TO LOCATION 10 IN
488						: USER MODE.
489	001504	010701			T12:	SCOPE
490	001506	012737	001542	000010	MOV	#T12A, @RESVEC
491	001514	005037	000012		CLR	@RESVEC+2
492	001520	012706	000500		MOV	#KPTR, KSP
493	001524	012737	140000	177776	MOV	#UM, @PSW ;USER MODE!!!
494	001532	000000			HALT	;HALT TRAPS IN USER MODE
495	001534	005037	177776		T12AA:	CLR @PSW
496	001540	000000			HALT	;ERROR! HALT DID NOT TRAP
497	001542	013700	177776		T12A:	MOV @PSW, R0
498	001546	005037	177776		CLR	@PSW
499	001552	022700	030000		CMP	#KM+PUM, R0
500	001556	001401			BEQ	.+4
501	001560	000000			HLT	
502	001562	022767	001534	176704	CMP	#T12AA, KPTR-4
503	001570	001401			BEQ	.+4

MO1

```

504 001572 000000          HLT
505
506          ;CHECK THAT SPL TRAPS TO 10 IN USER MODE.
507 001574 010701          †13: SCOPE
508 001576 012737 001626 000010      MOV      @T13A,@RESVEC
509 001604 012706 000500          MOV      @KPTR,KSP          ;SET KERNEL STACK PTR
510 001610 012737 140000 177776      MOV      @UM,@PSW          ;USER MODE!!!
511 001616 000237          SPL      7          ;SPL TRAPS IN USER MODE
512 001620 005037 177776      T13AA: CLR      @PSW          ;KERNEL MODE!!!
513 001624 000000          HLT          ;ERROR! SPL FAILED TO TRAP IN USER MODE
514 001626 013700 177776      T13A:  MOV      @PSW,R0
515 001632 005037 177776          CLR      @PSW
516 001636 022700 030000          CMP      @KM+PUM,R0
517 001642 001401          BEQ      .+4
518 001644 000000          HLT
519 001646 022767 001620 176620      CMP      @T13AA,KPTR-4
520 001654 001401          BEQ      .+4
521 001656 000000          HLT
522 001660 012737 000012 000010      MOV      @RESVEC+2,@RESVEC
523
524          ;TEST THAT "RESET" RESETS IN KERNEL MODE
525 001666 010701          †18: SCOPE
526 001670 005037 177776          CLR      @PSW
527 001674 012737 000340 177776      MOV      @PTY7,@PSW          ;PRIORITY TO 7
528 001702 012767 000100 175654      MOV      @100,177564        ;SET "IE" IN TPS
529 001710 000005          RESET          ;CLEAR "IE"
530 001712 005037 177776          CLR      @PSW
531 001716 032767 000100 175640      BIT      @100,177564
532 001724 001401          BEQ      .+4
533 001726 000000          HLT          ;RESET DID NOT
534          ;CLEAR "IE"
535
536          ;TEST THAT "RESET" NOP'S IN USER MODE
537 001730 010701          †19: SCOPE
538 001732 012737 140340 177776      MOV      @UM+PTY7,@PSW      ;USER MODE!!!
539 001740 012767 000100 175616      MOV      @100,177564        ;SET "IE"
540 001746 000005          RESET          ;SHOULD NOP
541 001750 032767 000100 175606      BIT      @100,177564
542 001756 001001          BNE      .+4
543 001760 000000          HLT          ;"IE" CLEARED
544 001762 005067 175576          CLR      177564
545 001766 005037 177776          CLR      @PSW
546
547          ;TEST INTERRUPT SEQUENCE USER TO KERNEL MODE
548 001772 010701          †15: SCOPE
549 001774 012706 000500          MOV      @KPTR,KSP          ;SET KERNEL STACK POINTER
550 002000 012737 170340 177776      MOV      @UM+PUM+PTY7,@PSW  ;USER MODE!!!
551 002006 012767 002052 176050      MOV      @T15A,64          ;INTERRUPT VEC.
552 002014 012767 000200 176044      MOV      @KM+PTY4,66
553 002022 012706 000700          MOV      @UPTR,USP          ;SET USER STACK POINTER
554 002026 042737 000200 177776      BIC      @PTY4,@PSW          ;SET PRIORITY LEVEL=3
555 002034 012767 000100 175522      MOV      @100,177564        ;REQUEST AN INTERRUPT AT LEVEL 4
556 002042 000240          NOP
557 002044 005037 177776      T15AA: CLR      @PSW          ;KERNEL MODE!!!
558 002050 000000          HLT          ;ERROR! NO INTERRUPT REQUEST
559 002052 013700 177776      T15A:  MOV      @PSW,R0          ;GET "NEW" @PSW
    
```

```

560 002056 005067 175502 CLR 177564 ;DISABLE REQUEST
561 002062 005037 177776 CLR 2#PSW
562 002066 022700 030200 CMP #KM+PUM+PTY4,R0 ;TEST THAT 'NEW' 2#PSW IS CORRECT
563 002072 001401 BEQ .+4 ;(PIRVEC+2)
564 002074 000000 HLT ;ERROR! 'NEW' 2#PSW NOT = TO (PIRVEC+2)
565 002076 022767 002044 176370 CMP #T15AA,KPTR-4 ;IS RETURN ADDRESS ON KERNEL STACK
566 002104 001401 BEQ .+4
567 002106 000000 HLT ;ERROR! RETURN ADDRESS NOT ON KERNEL STACK
568 002110 022767 170140 176360 CMP #UM+PUM+PTY3,KPTR-2 ;TEST THAT 'OLD' 2#PSW WAS SAVED ON
569 002116 001401 BEQ .+4 ;KERNEL STACK
570 002120 000000 HLT ;ERROR!
571 002122 012767 000066 175734 MOV #66,64
572 002130 005067 175732 CLR 66
573
574 ;TEST THAT THERE IS NO STACK OVERFLOW IN USER MODE.
575 002134 010701 T17: COPE
576 002136 012737 000400 177774 MOV #400,2#SLR ;SET STACK LIMIT =1000
577 002144 012737 140000 177776 MOV #UM,2#PSW ;USER MODE!!!
578 002152 012737 002402 000004 MOV #T17ERR,2#ERRVEC
579 002160 012706 000700 MOV #UPTR,USP ;SET USER STACK POINTER
580 002164 005067 176612 CLR TEMP ;CLEAR INDICATOR LOCATION
581 002170 004767 000006 T17A: JSR 7,T17B ;PUSH ONTO USER STACK
582 002174 052767 000400 176600 BIS #400,TEMP ;SET ERROR INDICATOR BIT
583 002202 052767 000001 176572 T17B: BIS #1,TEMP ;SET INDICATOR BIT
584 002210 004567 000006 JSR 5,T17C ;PUSH ONTO USER STACK
585 002214 052767 001000 176560 BIS #1000,TEMP ;SET ERROR INDICATOR BIT
586 002222 052767 000002 176552 T17C: BIS #2,TEMP ;SET INDICATOR BIT
587 002230 050546 BIS #R5,-(USP) ;PUSH ONTO USER STACK
588 002232 052767 000004 176542 BIS #4,TEMP ;SET INDICATOR BIT
589 002240 052737 000000 177776 BIS #REG,2#PSW ;SELECT R0-R5
590 002246 004767 000006 JSR 7,T17D ;PUSH ONTO USER STACK
591 002252 052767 002000 176522 BIS #2000,TEMP ;SET ERROR INDICATOR BIT
592 002260 052767 000010 176514 T17D: BIS #10,TEMP
593 002266 012702 002302 MOV #T17E,R2 ;SET UP RETURN FOR RTS
594 002272 000202 RTS R2 ;GO TO T17E
595 002274 052767 004000 176500 BIS #4000,TEMP ;SET INDICATOR TO SHOW ERROR
596 002302 052767 000020 176472 T17E: BIS #20,TEMP
597 002310 004567 000006 JSR R5,T17F ;SET ERROR INDICATOR BIT
598 002314 052767 010000 176460 BIS #10000,TEMP
599 002322 052767 000040 176452 T17F: BIS #40,TEMP
600 002330 012737 002354 000034 MOV #T17G,2#TRAPVEC ;SET UP TRAP VECTOR FOR TRAP
601 002336 012737 140000 000036 MOV #UM,2#TRAPVEC+2
602 002344 104400 TRAP
603 002346 052767 020000 176426 BIS #20000,TEMP
604 002354 052767 000100 176420 T17G: BIS #100,TEMP
605 002362 005037 177776 CLR 2#PSW ;KERNEL MODE!!!
606 002366 022767 000177 176406 CMP #177,TEMP
607 002374 001401 BEQ .+4
608 002376 000000 HLT
609 002400 000403 BR T17X
610 002402 005037 177776 T17ERR: CLR 2#PSW
611 002406 000000 HLT ;ERROR! OVERFLOW OCCURED
612 002410 005037 177774 T17X: CLR 2#SLR
613 002414 012737 000036 000034 MOV #TRAPVEC+2,2#TRAPVEC
614 002422 005067 175410 CLR TRAPVEC+2
615

```

```

616                                     ;TEST THAT MTPD/I POPS WORD OFF THE THE APPROPRIATE STACK (AS
617                                     ;DETERMINED BY BITS 15&14 IN @#PSW.)
618                                     ;MTPD, KERNEL MODE
619 002426 010701                                     †21: SCOPE
620 002430 005037 177776                               CLR @#PSW
621 002434 012706 000500                               MOV #KPTR,KSP ;SET KERNEL STACK POINTER
622 002440 012700 177777                               MOV #-1,R0 ;PRE-SET R0
623 002444 005016                                     CLR (KSP) ;PUT 0 ON THE STACK
624 002446 012737 030011 177776                       MOV #PUM+N+C,@#PSW ;PRE SET STATUS
625 002454 006600                                     MTP I R0 ;R0<--(KSP)+
626
627 002456 013702 177776                               MOV @#PSW,R2 ;GET STATUS
628 002462 022702 030005                               CMP #PUM+Z+C,R2
629 002466 001401                                     BEQ .+4
630 002470 000000                                     HLT ;ERROR! INCORRECT STATUS
631 002472 022706 000502                               CMP #KPTR+2,KSP ;DID KSP INCREMENT BY 2
632 002476 001401                                     BEQ .+4
633 002500 000000                                     HLT ;ERROR! KSP DID NOT POP
634 002502 005700                                     TST R0 ;DID WORD ON STACK (0) GET TO R0?
635 002504 001401                                     BEQ .+4
636 002506 000000                                     HLT ;ERROR! M72C DID NOT POP 0 OFF
637                                     ;KSP INTO R0
638
639                                     ;MTP I, KERNEL MODE
640 002510 010701                                     †22: SCOPE
641 002512 005037 177776                               CLR @#PSW
642 002516 012706 000500                               MOV #KPTR,KSP
643 002522 005002                                     CLR R2 ;PRESET R2
644 002524 012716 177777                               MOV #-1,(KSP)
645 002530 012737 030006 177776                       MOV #PUM+Z+V,@#PSW ;PRESET STATUS
646 002536 006602                                     MTP I R2 ;R2<-(KSP)+
647
648 002540 013700 177776                               MOV @#PSW,R0 ;GET STATUS
649 002544 022700 030010                               CMP #PUM+N,R0
650 002550 001401                                     BEQ .+4
651 002552 000000                                     HLT ;ERROR! INCORRECT STATUS
652 002554 022706 000502                               CMP #KPTR+2,KSP
653 002560 001401                                     BEQ .+4
654 002562 000000                                     HLT ;ERROR!
655 002564 005202                                     INC R2
656 002566 001401                                     BEQ .+4
657 002570 000000                                     HLT ;ERROR!
658
659                                     ;MTPD, USER MODE
660 002572 010701                                     †25: SCOPE
661 002574 012737 140000 177776                       MOV #UM,@#PSW
662 002602 012706 000700                               MOV #UPTR,USP
663 002606 052716 177777                               BIS #-1,(USP)
664 002612 000261                                     SEC
665 002614 042705 177777                               BIC #-1,R5
666 002620 006605                                     MTP I R5 ;R5<-(USP)+
667
668 002622 013700 177776                               MOV @#PSW,R0
669 002626 010602                                     MOV USP,R2
670 002630 005037 177776                               CLR @#PSW
671 002634 022700 140011                               CMP #UM+N+C,R0
    
```

672	002640	001401			BEQ	.+4	
673	002642	000000			HLT		
674	002644	022702	000702		CMP	#UPTR+2, R2	
675	002650	001401			BEQ	.+4	
676	002652	000000			HLT		
677	002654	005205			INC	R5	
678	002656	001401			BEQ	.+4	
679	002660	000000			HLT		
680							
681							
682							
683	002662	010701					
684	002664	012737	140000	177776			
685	002672	012706	000700				
686	002676	042716	177777				
687	002702	052700	177777				
688	002706	000257					
689	002710	006600					
690							
691	002712	013702	177776				
692	002716	010603					
693							
694	002720	005037	177776				
695	002724	022702	140004				
696	002730	001401					
697	002732	000000					
698	002734	022703	000702				
699	002740	001401					
700	002742	000000					
701	002744	005700					
702	002746	001401					
703	002750	000000					
704							
705							
706							
707							
708	002752	010701					
709	002754	012737	140000	177776			
710	002762	005006					
711	002764	012737	030000	177776			
712	002772	012706	000500				
713	002776	012716	000700				
714	013002	000277					
715	013004	006606					
716							
717	003006	013702	177776				
718	003012	012737	140000	177776			
719	003020	010600					
720	003022	005037	177776				
721	003026	022700	000700				
722	003032	001401					
723	003034	000000					
724	003036	022706	000502				
725	003042	001401					
726	003044	000000					
727							


```

;MTP1, USER MODE
+26: SCOPE
MOV #UM, @#PSW
MOV #UPTR, USP
BIC #-1, (USP)
BIS #-1, R0
CCC
MTP1 R0 ;R0+(USP)+

MOV @#PSW, R2
MOV USP, R3

CLR @#PSW
CMP #UM+2, R2
BEQ .+4
HLT
CMP #UPTR+2, R3
BEQ .+4
HLT
TST R0
BEQ .+4
HLT

;TEST THAT MTP D/I POPS WORD OFF STACK (AS DETERMINED BY BITS 15 & 14
;INTO STACK POINTER (AS DETERMINED BY BITS 13 & 12).
;USP+(KSP)+, MTPD
+30: SCOPE
MOV #UM, @#PSW ;USER MODE!!!
CLR USP ;PRESET USER STACK POINTER
MOV #KM+PUM, @#PSW ;KERNEL MODE!!! PREV USER MODE!!
MOV #KPTR, KSP ;SET KERNEL STACK POINTER
MOV #UPTR, (KSP)
SCC ;PRESET CC'S
MTP1 USP ;USP+(KSP)+

MOV @#PSW, R2 ;SAVE CC'S
MOV #UM, @#PSW ;USER MODE!!!
MOV USP, R0 ;GET USER STACK POINTER
CLR @#PSW ;KERNEL MODE!!!
CMP #UPTR, R0 ;CHECK THAT MTPD SET USER STACK
BEQ .+4 ;POINTER PROPERLY
HLT ;ERROR!
CMP #KPTR+2, KSP ;CHECK KERNEL STACK POINTER
BEQ .+4
HLT
    
```


728									
729									
730	003046	010701							
731	003050	012706	000500						
732	003054	012716	000736						
733	003060	006606							
734	003062	022706	000736						
735	003066	001401							
736	003070	000000							
737									
738									
739	003072	010701							
740	003074	012737	170000	177776					
741	003102	012706	000700						
742	003106	005016							
743	003110	000257							
744	003112	006606							
745									
746	003114	013700	177776						
747	003120	010602							
748	003122	005037	177776						
749	003126	022700	170004						
750	003132	001401							
751	003134	000000							
752	003136	005702							
753	003140	001401							
754	003142	000000							
755									
756									
757	003144	010701							
758	003146	012737	140000	177776					
759	003154	012706	177777						
760	003160	012737	030000	177776					
761	003166	005046							
762	003170	006606							
763									
764	003172	012737	140000	177776					
765	003200	010600							
766	003202	005037	177776						
767	003206	005700							
768	003210	001401							
769	003212	000000							
770									
771									
772	003214	010701							
773	003216	012737	170000	177776					
774	003224	012706	000700						
775	003230	012716	000700						
776	003234	006606							
777									
778	003236	010600							
779	003240	005037	177776						
780	003244	022700	000700						
781	003250	001401							
782	003252	000000							
783									

```

;KSP+(KSP)+ MTPD
↑31: SCOPE
      MOV      #KPTR,KSP
      MOV      #REDPTR,(KSP)
      MTPI     KSP          ;KSP+(KSP)+
      CMP      #REDPTR,KSP
      BEQ      .+4
      HLT

;:USP+(USP)+ MTPD
↑31C: SCOPE
      MOV      #UM+PUM,@#PSW ;USER MODE!!!, PREV USER MODE!!
      MOV      #UPTR,USP    ;SET USER STACK PTR
      CLR      (USP)        ;PUT #0 ON USER STACK
      CCC
      MTPI     USP          ;USP+(USP)+
      MOV      @#PSW,RO     ;SAVE CC'S
      MOV      USP,R2      ;SAVE USER STACK POINTER
      CLR      @#PSW       ;KERNEL MODE!!!
      CMP      #UM+PUM+Z,RO ;CHECK STATUS
      BEQ      .+4
      HLT                ;ERROR! INCORRECT STATUS AFTER MTPD
      TST      R2          ;CHECK NEW STACK POINTER VALUE
      BEQ      .+4
      HLT                ;ERROR! MTPD FAILED TO SET USER STACK POINTER

;USP+(KSP)+ MTPD
↑32A: SCOPE
      MOV      #UM,@#PSW    ;USER MODE
      MOV      #-1,USP      ;PRESET USER STACK POINTER
      MOV      #KM+PUM,@#PSW ;CURRENT KERNEL,PREVIOUS USER
      CLR      -(KSP)
      MTPI     USP          ;USP+(KSP)+
      MOV      #UM,@#PSW
      MOV      USP,RO      ;GET USER STACK POINTER
      CLR      @#PSW
      TST      RO
      BEQ      .+4
      HLT

;USP+(USP)+
↑35: SCOPE
      MOV      #UM+PUM,@#PSW
      MOV      #UPTR,USP
      MOV      #UPTR,(USP)
      MTPI     USP          ;USP+(USP)+
      MOV      USP,RO
      CLR      @#PSW
      CMP      #UPTR,RO
      BEQ      .+4
      HLT
    
```

```

784
785 ;TEST THAT MTPD/I TRAPS ON AN ODD ADDRESS DESTINATION
786 ;KERNEL MODE
787 003254 010701 T36: SCOPE
788 003256 005037 177776 CLR @#PSW
789 003262 012706 000500 MOV @KPTR,KSP
790 003266 012716 177777 MOV #-1,(KSP)
791 003272 012737 003312 000004 MOV @T36A,@#ERRVEC
792 003300 005067 174502 CLR ERRVEC+2
793 003304 006667 174467 MTPI -1 ;TRAPS ON ODD ADDRESS
794 003310 000000 T36AA: HLT ;ERROR! DID NOT TRAP
795 003312 022706 000476 T36A: CMP @KPTR-2,KSP ;IS KSP CORRECT?(1 POP AND 2
796 003316 001401 BEQ .+4 ;PUSHES)
797 003320 000000 HLT ;ERROR! INCORRECT VALUE IN KSP
798 003322 022767 003310 175146 CMP @T36AA,KPTR-2
799 003330 001401 BEQ .+4
800 003332 000000 HLT
801
802 ;USER MODE
803 003334 010701 T40: SCOPE
804 003336 012737 170000 177776 MOV @UM+PUM,@#PSW ;USER MODE!!!, PREV USER MODE!!
805 003344 012702 000001 MOV @1,R2
806 003350 012706 000700 MOV @UPTR,USP ;SET USER STACK POINTER
807 003354 012716 125252 MOV @125252,(USP) ;PRESET USER STACK
808 003360 012737 003404 000004 MOV @T40A,@#ERRVEC ;LOAD ERROR VECTOR
809 003366 012737 140000 000006 MOV @UM,@#ERRVEC+2
810 003374 006642 MTPI -(R2) ;-(R2)+(USP)+: SHOULD TRAP ON ODD ADRS
811 003376 005037 177776 T40AA: CLR @#PSW ;KERNEL MODE!!!
812 003402 000000 HLT ;ERROR DID NOT TRAP
813 003404 010600 T40A: MOV USP,R0 ;GET USERS STACK POINTER
814 003406 042737 140000 177776 BIC @UM,@#PSW ;KERNEL MODE!!!
815 003414 022700 000676 CMP @UPTR-2,R0 ;CHECK THAT USER STACK POINTER
816 003420 001401 BEQ .+4 ;PUSHED PROPERLY (1 POP 2 PUSHES)
817 003422 000000 HLT ;ERROR! INCORRECT USER STACK POINTER
818 003424 022737 170010 000700 CMP @UM+PUM+N,@#UPTR ;CHECK THAT CORRECT STATUS WAS
819 003432 001401 BEQ .+4 ;SAVED ON USER STACK ('N' IS DATA POPPED)
820 003434 000000 HLT ;ERROR! INCORRECT STATUS SAVED ON USER STACK
821 003436 022767 003376 175232 CMP @T40AA,UPTR-2 ;CHECK THAT RETURN ADDRESS WAS
822 003444 001401 BEQ .+4 ;SAVED ON USER STACK
823 003446 000000 HLT ;ERROR! RETURN PC NOT ON USER STACK
824 003450 022702 177777 CMP #-1,R2 ;DID R2 DECREMENT BY 2
825 003454 001401 BEQ .+4
826 003456 000000 HLT
827
828 ;TEST THAT MTP D/I CAN LOAD MEMORY ADDRESSES.
829 ;KERNEL MODE
830 003460 010701 T41: SCOPE
831 003462 005037 177776 CLR @#PSW
832 003466 012700 177777 MOV #-1,R0
833 003472 012737 003526 000004 MOV @T41A,@#ERRVEC
834 003500 005067 174302 CLR ERRVEC+2
835 003504 052737 000000 177776 BIS @REG,@#PSW ;R0-R5
836 003512 005000 CLR R0
837 003514 012746 000002 MOV @2,-(KSP)
838 003520 000261 SEC
839 003522 006620 MTPI (R0)+ ;(R0)+(KSP)+
839 003524 000401 BR .+4

```

```

840 003526 000000
841 003530 103401
842 003532 000000
843 003534 022767 000002 174236
844 003542 001401
845 003544 000000
846
847
848 003546 010701
849 003550 012737 003576 000004
850 003556 012706 000500
851 003562 012716 177777
852 003566 000257
853 003570 006637 001002
854
855 003574 000401
856 003576 000000
857 003600 013700 177776
858 003604 022700 000010
859 003610 001401
860 003612 000000
861 003614 005237 001002
862 003620 001401
863 003622 000000
864
865
866 003624 010701
867 003626 005037 177776
868 003632 012703 177777
869 003636 012737 003676 000004
870 003644 012737 140000 177776
871 003652 012703 001004
872 003656 005067 175120
873 003662 012706 000700
874 003666 052716 177777
875 003672 006643
876 003674 000401
877 003676 000000
878 003700 013700 177776
879 003704 042737 140000 177776
880 003712 122700 000010
881 003716 001401
882 003720 000000
883 003722 005167 175054
884 003726 001401
885 003730 000000
886 003732 012737 000006 000004
887 003740 005067 174042
888
889
890
891
892 003744 010701
893 003746 012706 000500
894 003752 012716 125252
895 003756 012700 177777
    
```

```

T41A:  HLT                ;ERROR! TRAPPED
        BCS                .+4        ;MTP D/I SHOULD NOT AFFECT CARRY
        HLT                ;BIT ERROR! CARRY BIT BUT CLEARED.
        CMP                #2,0
        BEQ                .+4
        HLT
T41B:  SCOPE
        MOV                #T41BB, @#ERRVEC ;LOAD ERROR VECTOR
        MOV                #KPTR, KSP      ;SET KERNEL STACK POINTER
        MOV                #-1, (KSP)     ;LOAD KERNEL STACK
        CCC                ;PRESET CC'S
        MTPI               @#TEMP        ;@#TEMP←(KSP)+
        BR                .+4
T41BB: HLT                ;ERROR! TRAPPED
        MOV                @#PSW, R0      ;SAVE CC'S
        CMP                #REG+N, R0    ;CHECK RESULT STATUS
        BEQ                .+4
        HLT                ;ERROR! INCORRECT STATUS AFTER MTPD
        INC                @#TEMP        ;CHECK RESULT
        BEQ                .-4
        HLT                ;ERROR! MTPD FAILED
    
```

```

;USER MODE
↑43:   SCOPE
        CLR                @#PSW
        MOV                #-1, R3
        MOV                #T43A, @#ERRVEC
        MOV                #UM, @#PSW
        MOV                @TEMP+2, R3
        CLR                TEMP
        MOV                #UPTR, USP
        BIS                #-1, (USP)
        MTPI               -(R3)        ;-(R3)←(USP)+
        BR                .+4
T43A:  HLT                ;ERROR TRAPPED
        MOV                @#PSW, R0
        BIC                #UM, @#PSW   ;KERNEL MODE!!!
        CMPB               #N, R0
        BEQ                .+4
        HLT
        COM                TEMP
        BEQ                .+4
        HLT
        MOV                #ERRVEC+2, @#ERRVEC
        CLR                ERRVEC+2
    
```

```

;TEST THAT MFP D/I PUSHES DESTINATION REGISTER DATA ONTO THE APPROPRIATE STACK
;(AS DETERMINED BY @#PSW BITS 15 & 14)
;KERNEL MODE, MFPD
↑44:   SCOPE
        MOV                #KPTR, KSP
        MOV                #125252, (KSP)
        MOV                #-1, R0
    
```

896	003762	000261		SEC			
897	003764	006500		MFPI	R0	;	-(KSP)+R0, (R0)=-1
898	003766	013702	177776	MOV	@#PSW, R2	;	GET STATUS RESULT
899	003772	022702	000011	CMP	#REG+N+C, R2	;	
900	003776	001401		BEQ	.+4	;	
901	004000	000000		HLT		;	ERROR! INCORRECT STATUS RESULT
902	004002	022706	000476	CMP	#KPTR-2, KSP	;	DID KERNEL STACK POINTER GET
903	004006	001401		BEQ	.+4	;	PUSHED?
904	004010	000000		HLT		;	ERROR!
905	004012	005116		COM	(KSP)	;	TEST THAT CORRECT DATA(-1) GOT
906	004014	001401		BEQ	.+4	;	PUSHED ONTO KERNEL STACK
907	004016	000000		HLT		;	ERROR! -1 NOT PUSHED ONTO KERNEL STACK
908				;			
909	004020	010701		;			
910	004022	012706	000500	↑45: KERNEL MODE, MFPI	SCOPE		
911	004026	012716	052525	MOV	#KPTR, KSP	;	PRE SET STACK
912	004032	005004		MOV	#52525, (KSP)	;	PRESET 'WRONG' REGISTER
913	004034	012737	000001	CLR	R4	;	SELECT R0-R5, SET C
914	004042	012704	125252	MOV	#REG+C, @#PSW	;	LOAD DATA TO BE MOVED
915	004046	006504		MOV	#125252, R4	;	-(KSP)+R4, (R4)=125252
916				MFPI	R4	;	
917	004050	013700	177776	MOV	@#PSW, R0		
918	004054	022700	000011	CMP	#REG+N+C, R0	;	CHECK STATUS RESULT
919	004060	001401		BEQ	.+4	;	
920	004062	000000		HLT		;	ERROR! INCORRECT STATUS
921	004064	022706	000476	CMP	#KPTR-2, KSP	;	CHECK PUSH
922	004070	001401		BEQ	.+4	;	
923	004072	000000		HLT		;	ERROR! KSP DID NOT PUSH DOWN
924	004074	022716	125252	CMP	#125252, (KSP)	;	CHECK DATA ON THE STACK
925	004100	001401		BEQ	.+4	;	
926	004102	000000		HLT		;	ERROR! INCORRECT DATA ON THE STACK
927				;		;	IF DATA=0 THEN INCORRECT REGISTER
928				;		;	(R4), IF DATA=52525 NO DATA PUSHED
929				;		;	ON THE STACK.
930				;			
931	004104	010701		↑50: USER MODE, MFPD	SCOPE		
932	004106	005003		CLR	R3	;	PRESET
933	004110	012737	140000	MOV	#UM, @#PSW	;	USER MODE, R0-R5
934	004116	012706	000700	MOV	#UPTR, USP	;	SET USER'S STACK POINTER
935	004122	012726	125252	MOV	#125252, (USP)+	;	PRESET STACK
936	004126	012703	177777	MOV	#-1, R3	;	
937	004132	000257		CCC		;	
938	004134	006503		MFPI	R3	;	-(USP)+R3 (R3)=-1
939							
940	004136	013700	177776	MOV	@#PSW, R0		
941	004142	010604		MOV	USP, R4		
942	004144	042737	140000	BIC	#UM, @#PSW		
943	004152	022700	140010	CMP	#UM+N, R0		
944	004156	001401		BEQ	.+4		
945	004160	000000		HLT			
946	004162	022704	000700	CMP	#UPTR, R4		
947	004166	001401		BEQ	.+4		
948	004170	000000		HLT			
949	004172	005214		INC	(R4)		
950	004174	001401		BEQ	.+4		
951	004176	000000		HLT			

Address	Instruction	OpCode	PC	PSW	Comments
952	CLR	005037	177776	2#PSW	
953					
954	USER MODE MFPI	010701			↑51: SCOPE
955	CLR	005005		R5	
956	MOV	012737	140000	2#UM, 2#PSW	; USER MODE!!!
957	MOV	012706	000700	2#UPTR, USP	; SET USER STACK POINTER
958	MOV	012716	177777	2#-1, (USP)	; PRESET USER STACK
959	MOV	012705	000700	2#UPTR, R5	; PRESET R5
960	SCC	000277			; PRESET CONDITION CODES
961	MFPI	006505		R5	; -(USP)+R5
962					
963	MOV	013700	177776	2#PSW, R0	; GET STATUS RESULT
964	MOV	010602		USP, R2	; GET USER STACK POINTER
965	BIC	042737	140000	2#UM, 2#PSW	; KERNEL MODE!!!
966	CMP	022700	140001	2#UM+2, R0	; CHECK STATUS RESULT AFTER MFPI INST
967	BEQ	001401		.+4	
968	HLT	000000			; ERROR! INCORRECT STATUS AFTER MFPI
969	CMP	022702	000676	2#UPTR-2, R2	
970	BEQ	001401		.+4	
971	HLT	000000			
972	CMP	022712	000700	2#UPTR, (R2)	
973	BEQ	001401		.+4	
974	HLT	000000			
975					; TEST THAT MFPD/I PUSHES DESTINATION MEMORY DATA ONTO THE APPROPRIATE
976					; STACK.
977					; KERNEL MODE, MFPD
978	USER MODE MFPI	010701			↑52: SCOPE
979	CLR	005037	177776	2#PSW	; KERNEL MODE!!!
980	MOV	012700	001002	2#TEMP, R0	; PRESET R0
981	BIS	052737	000000	2#REG, 2#PSW	; SELECT R0-R5
982	MOV	012700	001004	2#TEMP+2, R0	; PRESET R0
983	MOV	012767	177777	2#-1, TEMP	
984	CLR	005067	174444	TEMP+2	
985	MOV	012706	000500	2#KPTR, KSP	; SET KERNEL STACK POINTER
986	MOV	012716	125252	2#125252, (KSP)	; PRESET KERNEL STACK
987	MFPI	006520		(R0)+	; -(KSP)+(R0)+, R0=TEMP+2, TEMP+2=0
988					
989	MOV	013702	177776	2#PSW, R2	
990	CMP	022702	000004	2#REG+2, R2	
991	BEQ	001401		.+4	
992	HLT	000000			
993	CMP	022706	000476	2#KPTR-2, KSP	
994	BEQ	001401		.+4	
995	HLT	000000			
996	TST	005716		(KSP)	
997	BEQ	001401		.+4	
998	HLT	000000			
999					
1000					
1001	USER MODE MFPI	010701			↑54: SCOPE
1002	MOV	012737	140000	2#UM, 2#PSW	
1003	MOV	012703	001004	2#TEMP+2, R3	
1004	BIS	052737	000340	2#REG+PRTY7, 2#PSW	
1005	MOV	012703	001006	2#TEMP+4, R3	
1006	CLR	005067	174344	TEMP	
1007	MOV	012767	177777	2#-1, TEMP+2	

1008	004444	012706	000700		MOV	#UPTR, USP	
1009	004450	012716	125252		MOV	#125252, (USP)	
1010	004454	006563	177776		MFPI	-2(R3)	; -(USP+-2(R3), R3=#TEMP+4, TEMP+2=-1
1011							
1012	004460	013700	177776		MOV	#PSW, R0	
1013	004464	010602			MOV	USP, R2	
1014	004466	042737	140000	177776	BIC	#UM, #PSW	
1015	004474	022700	140350		JMP	#UM+PRTY7+N, R0	
1016	004500	001401			BEQ	.+4	
1017	004502	000000			HLT		
1018	004504	022702	000676		CMP	#UPTR-2, R2	
1019	004510	001401			BEQ	.+4	
1020	004512	000000			HLT		
1021	004514	005112			COM	(R2)	
1022	004516	001401			BEQ	.+4	
1023	004520	000000			HLT		
1024							
1025	004522	010701					
1026	004524	012737	030000	177775	↑55: SCOPE		
1027	004532	012706	001000		MOV	#PUM, #PSW	; KERNEL MODE!!!, PREV USER MODE!!
1028	004536	012767	177777	174236	MOV	#YELPTR, KSP	; SET STACK PTR AT TOP OF YELLOW ZONE
1029	004544	005066	177776		MOV	#-1, TEMP	; PRESET DATA
1030	004550	012737	004576	000004	CLR	-2(KSP)	; PRESET STACK DATA
1031	004556	005037	000006		MOV	#T55A, #ERRVEC	; LOAD ERROR TRAP VECTOR
1032	004562	012737	000400	177774	CLR	#ERRVEC+2	
1033	004570	006567	174206		MOV	#400, #SLR	; SET STACK LIMIT =1000
1034					MFPI	TEMP	; PUSH TEMP ONTO KERNEL STACK
1035	004574	000000					; SHOULD OVERFLOW STACK
1036	004576	022767	177777	174172	T55AA: HLT		; ERROR! FAILED TO TRAP ON OVERFLOW
1037	004604	001401			T55A: CMP	#-1, YELPTR-2	; CHECK THAT MFPI PUSHED DATA
1038	004606	000000			BEQ	.+4	; ONTO STACK
1039	004610	022767	030010	174156	HLT		; ERROR! MFPI FAILED TO PUSH DATA
1040	004616	001401			CMP	#PUM+N, YELPTR-4	; CHECK SAVED STATUS ON TRAP
1041	004620	000000			BEQ	.+4	
1042	004622	022767	004574	174142	HLT		; ERROR! INCORRECT STATUS SAVED
1043	004630	001401			CMP	#T55AA, YELPTR-6	; CHECK SAVED PC ON STACK
1044	004632	000000			BEQ	.+4	
1045	004634	005037	177774		HLT		; ERROR! INCORRECT PC SAVED ON STACK
1046					CLR	#SLR	; CLEAR STACK LIMIT REGISTER
1047							
1048	004640	010701					
1049	004642	012737	004714	000004	↑56: SCOPE		
1050	004650	012737	030340	177776	MOV	#T56A, #ERRVEC	; SET ERROR TRAP VECTOR
1051	004656	012706	000736		MOV	#PUM+PRTY7, #PSW	; KERNEL MODE!!!, PREV USER MODE!!
1052	004662	012766	177777	177776	MOV	#REDPTR, KSP	; SET STACK PTR TO TOP OF RED ZONE
1053	004670	005067	174106		MOV	#-1, -2(KSP)	; PRESET RED LOCATION=-1
1054					CLR	TEMP	; (TEMP) WILL BE THE DATA MOVED
1055	004674	012703	001004				; TO RED LOCATION
1056	004700	012737	000400	177774	MOV	#TEMP+2, R3	; LOAD INDEX REGISTER
1057	004706	006563	177776		MOV	#400, #SLR	; SET STACK LIMIT=1000
1058					MFPI	-2(R3)	; -(KSP)+TEMP SHOULD OVER
1059	004712	000000					; FLOW (RED)
1060							; ERROR! FAILED TO TRAP ON 'RED'
1061	004714	022737	177777	000734	T56AA: HLT		; OVERFLOW
1062	004722	001401			T56A: CMP	#-1, #REDPTR-2	; TEST THAT MFPI DID NOT WRITE
1063	004724	000000			BEQ	.+4	; INTO 'RED' LOCATION
					HLT		; ERROR!

1064	004726	005706			TST	KSP		; STACK SHOULD HAVE GONE TO 0
1065	004730	001401			BEQ	.+4		
1066	004732	000000			HLT			
1067	004734	022737	030344	000002	CMP	#PUM+PRTY7+Z, #2		; OLD STATUS SHOULD BE IN 2
1068	004742	001401			BEQ	.+4		
1069	004744	000000			HLT			; ERROR!
1070	004746	022737	004712	000000	CMP	#T56AA, #0		; AND RETURN IN 0
1071	004754	001401			BEQ	.+4		
1072	004756	000000			HLT			; ERROR! INCORRECT PC IN 0
1073	004760	005037	177774		CLR	#SLR		
1074	004764	012737	000006	000004	MOV	#ERRVEC+2, #ERRVEC		; RESTORE ERROR VECTOR
1075								
1076								; TEST TRAP & RETURN USER-KERNEL-USER
1077	004772	010701			T57:	SCOPE		
1078	004774	012706	000500		MOV	#KPTR, KSP		; SET KERNEL STACK POINTER
1079	005000	012737	000340	000036	MOV	#PRTY7, #TRAPVEC+2		
1080	005006	012737	005076	000034	MOV	#T57A, #TRAPVEC		
1081	005014	012737	140000	177776	MOV	#UM, #PSW		; USER MODE!!!
1082	005022	005002			CLR	R2		
1083	005024	104400			TRAP			; TRAP & ENTER KERNEL MODE
1084	005026	013767	177776	173746	T57AA:	MOV	#PSW, TEMP	
1085	005034	042737	140000	177776	BIC	#UM, #PSW		; KERNEL MODE!!!
1086	005042	022767	005026	173424	CMP	#T57AA, KPTR-4		; CHECK THAT RETURN ADDRESS IS ON
1087	005050	001401			BEQ	.+4		; KERNEL STACK
1088	005052	000000			HLT			; ERROR! RETURN ADDRESS NOT ON STACK
1089	005054	022767	140004	173720	CMP	#UM+Z, TEMP		; CHECK THAT CORRECT #PSW WAS
1090	005062	001401			BEQ	.+4		; RESTORED ON THE RETURN
1091	005064	000000			HLT			; ERROR! INCORRECT STATUS WAS RETURNED
1092								; BY KERNEL FROM TRAP
1093	005066	005102			COM	R2		; CHECK THAT TRAP ROUTINE WAS EXECUTED
1094	005070	001401			BEQ	.+4		
1095	005072	000000			HLT			; ERROR! KERNEL DID NOT DO COM R2
1096								; (AT T57A)
1097	005074	000402			T57A:	BR	T57EX	; EXIT TEST
1098	005076	005102			COM	R2		; COMPLEMENT R2
1099	005100	000002			RTI			; AND EXIT
1100	005102	000240			T57EX:	NOP		
1101								
1102								; TEST THAT MFPD/I CAN PUSH ONTO CURRENT STACK (AS DETERMINED BY PS15 &
1103								; PS14) THE PREVIOUS MODES STACK POINTER (AS DETERMINED BY PS13 & PS12)
1104								; -(KSP)+KSP, MFPD
1105	005104	010701			T60:	SCOPE		
1106	005106	005037	177776		CLR	#PSW		; KERNEL MODE!!!, PREV KERNEL MODE!!
1107	005112	012706	000500		MOV	#KPTR, KSP		; SET KERNEL STACK POINTER
1108	005116	006506			MFPI	KSP		; -(KSP)+KSP
1109	005120	022767	000500	173350	CMP	#KPTR, KPTR-2		; TEST THAT VALUE OF KERNEL STACK POINTER
1110	005126	001401			BEQ	.+4		; WAS PUSHED ONTO KERNEL STACK
1111	005130	000000			HLT			; ERROR!
1112								
1113								
1114	005132	010701			T62:	-(KSP)+USP, MFPD		
1115	005134	012737	030000	177776	SCOPE			
1116	005142	012706	000500		MOV	#KM+PUM, #PSW		; KERNEL MODE!!!, PREV USER MODE!!
1117	005146	012716	177777		MOV	#KPTR, KSP		; SET KERNEL STACK POINTER
1118	005152	006606			MOV	#-1, (KSP)		
1119	005154	005166	177776		MTPU	USP		; SET USER STACK POINTER USP+(KSP)+
					COM	-2(KSP)		; PRESET KERNEL STACK

1120	005160	006506			MFPI	USP	;- (KSP)+USP
1121	005162	022716	177777		CMP	#-1, (KSP)	;CHECK THAT USER STACK POINTER WAS
1122	005166	001401			BEQ	.+4	;PUSHED ONTO KERNEL STACK
1123	005170	000000			HLT		;ERROR!
1124							
1125							
1126	005172	010701					
1127	005174	012737	030000	177776	↑65:	SCOPE	
1128	005202	012706	000500		MOV	#PUM, @#PSW	;KERNEL MODE!!!, PREV USER MODE!!
1129	005206	012716	000700		MOV	#KPTR, KSP	;SET KERNEL STACK POINTER
1130	005212	006606			MOV	#UPTR, (KSP)	
1131	005214	005067	173456		MTPI	USP	;SET USER STACK POINTER
1132	005220	052737	140000	177776	CLR	UPTR-2	
1133	005226	006506			BIS	#UM, @#PSW	;USER MODE!!!, PREV USER MODE!!!
1134	005230	042737	140000	177776	MFPI	USP	;PUSH USER STACK POINTER ONTO USER STACK
1135	005236	006506			BIC	#UM, @#PSW	;KERNEL MODE!!!, PREV USER MODE!!
1136	005240	022716	000676		MFPI	USP	;PUSH USER STACK POINTER ONTO KERNEL STACK
1137	005244	001401			CMP	#UPTR-2, (KSP)	;CHECK THAT USER STACK POINTER WAS
1138	005246	000000			BEQ	.+4	;PUSHED PROPERLY (ONCE)
1139	005250	022767	000700	173420	HLT		;ERROR!
1140	005256	001401			CMP	#UPTR, UPTR-2	;CHECK THAT USER STACK POINTER IS ON THE
1141	005260	000000			BEQ	.+4	;USERS STACK
1142					HLT		;ERROR!
1143							
1144	005262	010701					
1145	005264	005037	177776		↑66:	SCOPE	
1146	005270	012706	000500		CLR	@#PSW	;KERNEL MODE!!!, PREV KERNEL MODE!!
1147	005274	006506			MOV	#KPTR, KSP	;SET KERNEL STACK POINTER
1148					MFPI	KSP	;PUSH KERNEL STACK POINTER ONTO KERNEL
1149	005276	022767	000500	173172			;STACK
1150	005304	001401			CMP	#KPTR, KPTR-2	;CHECK RESULT
1151	005306	000000			BEQ	.+4	
1152					HLT		;ERROR!
1153							
1154	005310	010701					
1155	005312	012737	030000	177776	↑70:	SCOPE	
1156	005320	012706	000500		MOV	#PUM, @#PSW	;KERNEL MODE!!!, PREV USER MODE!!
1157	005324	012716	177777		MOV	#KPTR, KSP	;SET KERNEL STACK POINTER
1158	005330	006606			MOV	#-1, (KSP)	
1159	005332	005166	177776		MTPI	USP	;SET USER STACK POINTER
1160	005336	006506			COM	-2(KSP)	;PRESET KERNEL STACK
1161	005340	022716	177777		MFPI	USP	;PUSH USER STACK POINTER ONTO KERNEL STACK
1162	005344	001401			CMP	#-1, (KSP)	;CHECK RESULT
1163	005346	000000			BEQ	.+4	
1164					HLT		;ERROR! USER STACK POINTER NOT ON KERNEL STACK
1165							
1166	005350	010701					
1167	005352	012737	030000	177776	↑73:	SCOPE	
1168	005360	012706	000500		MOV	#PUM, @#PSW	;KERNEL MODE!!!, PREV USER MODE!!
1169	005364	012716	000700		MOV	#KPTR, KSP	;SET KERNEL STACK POINTER
1170	005370	006606			MOV	#UPTR, (KSP)	
1171	005372	005067	173300		MTPI	USP	;SET USER STACK POINTER
1172	005376	052737	140000	177776	CLR	UPTR-2	;PRESET USER STACK
1173	005404	006506			BIS	#UM, @#PSW	;USER MODE!!!, PREV USER MODE!!
1174	005406	042737	140000	177776	MFPI	USP	;- (USP)+USP
1175	005414	006506			BIC	#UM, @#PSW	;KERNEL MODE!!!
					MFPI	USP	;GET USER STACK POINTER


```

1176 005416 022716 000676      CMP      #UPTR-2,(KSP)  ;CHECK THAT USER STACK PCINTER WAS
1177 005422 001401      BEQ      .+4          ;PUSHED ONCE
1178 005424 000000      HLT                      ;ERROR!
1179 005426 022767 000700 173242    CMP      #UPTR,UPTR-2 ;CHECK THAT USER STACK POINTER WAS PUSHED
1180 005434 001401      BEQ      .+4          ;ONTO USER STACK
1181 005436 000000      HLT                      ;ERROR!
1182
1183      ;TEST THAT ILLEGAL MODE DOES NOT HANG BUS.
1184      †74:  SCOPE
1185 005440 010701      MOV      #IM,#PSW      ;ILLEGAL MODE!!!
1186 005442 012737 100000 177776    MOV      #PSW,R0      ;GET ILLEGAL MODE
1187 005450 013700 177776    MOV      #PSW,R0      ;GET ILLEGAL MODE
1188 005454 005037 177776    CLR      #PSW          ;KERNEL MODE!!
1189 005460 022700 100000    CMP      #IM,R0        ;CHECK THAT ILLEGAL MODE WAS SET
1190 005464 001401      BEQ      .+4          ;INTO STATUS
1191 005466 000000      HLT
1192
1193      ;TEST THAT ILLEGAL MODE DOES NOT HANG BUS.
1194      †75:  SCOPE
1195 005470 010701      MOV      #IM1,#PSW    ;ILLEGAL MODE!!!
1196 005472 012737 040000 177776    MOV      #PSW,R0      ;GET ILLEGAL MODE
1197 005500 013700 177776    MOV      #PSW,R0      ;GET ILLEGAL MODE
1198 005504 005037 177776    CLR      #PSW          ;KERNEL MODE!!
1199 005510 022700 040000    CMP      #IM1,R0      ;CHECK THAT ILLEGAL MODE WAS SET
1200 005514 001401      BEQ      .+4          ;INTO STATUS
1201 005516 000000      HLT
1202
1203      ;TEST THAT KERNEL CAN GET DATA FROM USER STACK
1204      †76:  SCOPE
1205 005520 010701      MOV      #KM+PUM,#PSW ;KERNEL MODE!!!, PREV USER MODE!!
1206 005522 012737 030000 177776    MOV      #KPTR,KSP    ;SET KERNEL STACK POINTER
1207 005530 012706 000500      MOV      #UPTR,(KSP) ;SET USER STACK POINTER
1208 005534 012716 000700      MTPI    USP           ;PRESET USER STACK
1209 005540 006606      CLR      USP         ;PRESET KERNEL STACK
1210 005542 005067 173132      CLR      (KSP)
1211 005546 005016      MOV      #-1,-2(KSP) ;-(KSP)+USP
1212 005550 012766 177777 177776    MFPI    USP           ;LIKE MOV 2(6),-(6)
1213 005554 006506      MFPI    2(KSP)
1214 005556 000240      NOP
1215 005560 013703 177776      MOV      #PSW,R3      ;SAVE STATUS RESULT
1216 005564 022767 000700 172700    CMP      #UPTR,KPTR   ;CHECK THAT USER STACK POINTER WAS
1217 005572 001401      BEQ      .+4          ;PUSHED ONTO KERNEL STACK
1218 005576 000000      HLT                      ;ERROR!
1219 005580 022706 000476      CMP      #KPTR-2,KSP  ;CHECK THAT KERNEL STACK POINTER IS POS-
1220 005584 001401      BEQ      .+4          ;ITIONED PROPERLY
1221 005588 000000      HLT                      ;ERROR! INCORRECT KERNEL STACK POINTER
1222 005592 005716      TST      (KSP)        ;CHECK THAT CORRECT DATA
1223 005596 001401      BEQ      .+4          ;WAS PUSHED ONTO KERNEL STACK
1224 005600 000000      HLT                      ;ERROR!
1225 005604 022703 030004      CMP      #KM+PUM+2,R3 ;CHECK STATUS
1226 005608 001401      BEQ      .+4
1227 005612 000000      HLT                      ;ERROR! INCORRECT STATUS
1228
1229      ;CHECK THAT MTPD CAN LOAD MEMORY ADDRESS DM=7,PC
1230      †102: SCOPE
1231 005632 010701      MOV      #KM+PUM,#PSW ;KERNEL MODE!!!, PREV USER MODE!!
1232 005634 012737 030000 177776    MOV      #KPTR,KSP    ;SET KERNEL STACK PTR
1233 005642 012706 000500      CLR      (KSP)        ;PUT DATA ON STACK
1234 005646 005016

```

1232	005650	012737	001002	001004	MOV	#TEMP, #TEMP+2	; LOAD ADDRESS
1233	005656	012767	177777	173116	MOV	#-1, TEMP	; PRESET DATA
1234	005664	000277			SCC		; PRESET CC'S
1235	005666	006677	173112		MTPi	#TEMP+2	; TEMP+(KSP)+
1236	005672	013703	177776		MOV	#PSW, R3	; CHECK CC'S
1237	005676	022703	030005		CMP	#PUM+2+C, R3	; CHECK CC'S
1238	005702	001401			BEQ	.+4	
1239	005704	000000			HLT		; ERROR! INCORRECT CC'S AFTER MTPD
1240	005706	005737	001002		TST	#TEMP	; CHECK RESULT
1241	005712	001401			BEQ	.+4	
1242	005714	000000			HLT		; ERROR! INCORRECT RESULT
1243							
1244							; CHECK THAT MTPi CAN LOAD MEMORY ADDRESS DM=7
1245	005716	010701			T103: SCOPE		
1246	005720	012737	030000	177776	MOV	#KM+PUM, #PSW	; KERNEL MODE!!!
1247	005726	012706	000500		MOV	#KPTR, KSP	; SET KERNEL STACK PTR
1248	005732	012716	177777		MOV	#-1, (KSP)	; LOAD DATA ONTO STACK
1249	005736	012704	177776		MOV	#-2, R4	; LOAD INDEX REGISTER
1250	005742	005067	173034		CLR	TEMP	; PRESET DATA
1251	005746	012767	001002	173030	MOV	#TEMP, TEMP+2	
1252	005754	006674	001006		MTPi	#TEMP+4(R4)	; TEMP+(KSP)+
1253	005760	013703	177776		MOV	#PSW, R3	; SAVE STATUS RESULT
1254	005764	022706	000502		CMP	#KPTR+2, KSP	; CHECK THAT KSP POPPED
1255	005770	001401			BEQ	.+4	
1256	005772	000000			HLT		; ERROR! INCORRECT STACK PTR
1257	005774	022703	030010		CMP	#PUM+N, R3	; CHECK STATUS RESULT
1258	006000	001401			BEQ	.+4	
1259	006002	000000			HLT		; ERROR! INCORRECT STATUS
1260	006004	005267	172772		INC	TEMP	; CHECK RESULT
1261	006010	001401			BEQ	.+4	
1262	006012	000000			HLT		; ERROR! INCORRECT RESULT
1263							
1264							; TEST THAT MTPD/I CAN LOAD PC
1265	006014	010701			T104: SCOPE		
1266	006016	012737	000000	177776	MOV	#KM, #PSW	; KERNEL MODE!!!
1267	006024	012706	000500		MOV	#KPTR, KSP	; SET KERNEL STACK PTR
1268	006030	012716	006042		MOV	#T104A, (KSP)	; PUT NEW PC ON STACK
1269	006034	000277			SCC		; PRESET CC'S
1270	006036	006607			MTPi	PC	; PC+(KSP)+
1271	006040	000000			HLT		; ERROR! MTPD FAILED TO SET PC
1272	006042	100001			T104A: BPL	.+4	
1273	006044	000000			HLT		; ERROR! 'N' FAILED TO CLEAR IN STATUS
1274	006046	103401			BCS	.+4	
1275	006050	000000			HLT		; ERROR! 'C' WAS CLEARED BY MTPD
1276							
1277							; USER MODE
1278	006052	010701			T106: SCOPE		
1279	006054	012737	170000	177776	MOV	#UM+PUM, #PSW	; USER MODE!!!
1280	006062	012706	000700		MOV	#UPTR, USP	; SET USER STACK PTR
1281	006066	012716	006104		MOV	#T106A, (USP)	; PUT NEW PC ON STACK
1282	006072	000277			SCC		; PRESET CC'S
1283	006074	006607			MTPi	PC	; PC+(USP)+
1284	006076	005037	177776		CLR	#PSW	; KERNEL MODE!!!
1285	006102	000000			HLT		; ERROR! MTPD FAILED TO LOAD PC
1286	006104	013705	177776		T106A: MOV	#PSW, R5	; SAVE STATUS
1287	006110	005037	177776		CLR	#PSW	; KERNEL MODE!!!

1288	006114	022705	170001		CMP	#UM+PUM+C,R5	;CHECK STATUS
1289	006120	001401			BEQ	.+4	
1290	006122	000000			HLT		
1291							
1292							
1293	006124	010701					
1294	006126	005037	177776		↑107: CLR	@#PSW	;KERNEL MODE!!!
1295	006132	012706	000500		MOV	#KPTR,KSP	;SET KERNEL STACK PTR
1296	006136	012737	006154	000004	MOV	#T107A,@#ERRVEC	;LOAD ERROR VECTOR
1297	006144	000277			SCC		;PRESET CC'S
1298	006146	006567	171627		MFPI	1	;ODD ADDRESS SHOULD TRAP
1299	006152	000000			T107AA: HLT		;ERROR! FAILED TO TRAP ON ODD ADDRESS
1300	006154	022706	000474		T107A: CMP	#KPTR-4,KSP	;CHECK THAT STACK PTR WAS PUSHED
1301	006160	001401			BEQ	.+4	;PROPERLY (2 PUSHES)
1302	006162	000000			HLT		;ERROR! INCORRECT STACK PTR AFTEP ERROR
1303	006164	022726	006152		CMP	#T107AA,(KSP)+	;CHECK RETURN PC ON STACK
1304	006170	001401			BEQ	.+4	
1305	006172	000000			HLT		;ERROR! RETURN PC NOT ON STACK
1306	006174	022716	000017		CMP	#17,(KSP)	;CHECK SAVED STATUS ON STACK
1307	006200	001401			BEQ	.+4	
1308	006202	000000			HLT		;ERROR! INCORRECT STATUS SAVED ON STACK
1309							
1310	006204	010701			↑110: USER MODE, TIME OUT		
1311	006206	012737	140000	177776	↑110: SCOPE		
1312	006214	012706	000700		MOV	#UM,@#PSW	;USER MODE!!!
1313	006220	012737	140000	000006	MOV	#UPTR,USP	;SET USER STACK
1314	006226	012737	006246	000004	MOV	#UM,@#ERRVEC+2	;LOAD 'NEW' STATUS
1315	006234	006537	177702		MOV	#T110A,@#ERRVEC	;AND PC
1316	006240	005037	177776		MFPI	@#177702	;177702 IS NON-EXISTANT ADRS
1317	006244	000000			T110AA: CLR	@#PSW	;KERNEL MODE!!!
1318	006246	010603			HLT		;ERROR! DID NOT TRAP ON NON ADRS
1319	006250	042737	140000	177776	T110A: MOV	USP,R3	;SAVE USER STACK PTR
1320	006256	022703	000674		BIC	#UM,@#PSW	;KERNEL MODE!!!
1321	006262	001401			CMP	#UPTR-4,R3	;CHECK USER STACK PTR
1322	006264	000000			BEQ	.+4	
1323	006266	022723	006240		HLT		;ERROR! INCORRECT USP AFTER ERROR TRAP
1324	006272	001401			CMP	#T110AA,(R3)+	;CHECK RETURN PC ON USER STACK
1325	006274	000000			BEQ	.+4	
1326	006276	022713	140000		HLT		;ERROR! RETURN PC NOT ON USER STACK
1327	006302	001401			CMP	#UM,(R3)	;CHECK SAVED STATUS
1328	006304	000000			BEQ	.+4	
1329					HLT		;ERROR! INCORRECT STATUS SAVED ON STACK
1330							
1331	006306	010701			↑111: USER MODE, ODD ADDRESS		
1332	006310	012737	140000	177776	↑111: SCOPE		
1333	006316	012706	000700		MOV	#UM,@#PSW	;USER MODE!!!
1334	006322	012737	006350	000004	MOV	#UPTR,USP	;SET USER STACK PTR
1335	006330	012737	140000	000006	MOV	#T111A,@#ERRVEC	;LOAD ERROR TRAP VECTOR
1336	006336	006567	171435		MOV	#UM,@#ERRVEC+2	
1337	006342	005037	177776		MFPI	-1	;ODD ADDRESS SHOULD TRAP
1338	006346	000000			T111AA: CLR	@#PSW	;KERNEL MODE!!!
1339	006350	010603			HLT		;ERROR! FAILED TO TRAP
1340	006352	042737	140000	177776	T111A: MOV	USP,R3	;SAVE USER STACK PTR
1341	006360	022703	000674		BIC	#UM,@#PSW	;KERNEL MODE!!!
1342	006364	001401			CMP	#UPTR-4,R3	;CHECK USER STACK PTR
1343	006366	000000			BEQ	.+4	
					HLT		;ERROR! INCORRECT USER STACK POINTER

```

1344 006370 022713 006342      CMP      #T111AA,(R3)      ;CHECK RETURN SDDRESS ON USER STACK
1345 006374 001401      BEQ      .+4
1346 006376 000000      HLT
1347 006400 012737 000006 000004      MOV      #ERRVEC+2,#ERRVEC;RESTORE ERROR TRAP TO HALT
1348 006406 005067 171374      CLR      ERRVEC+2
1349
1350
1351      ;TEST THAT MTPD INSTRUCTION CAN LOAD DATA TO AN ADDRESS VIA THE STACK
1352      ;KERNEL MODE,PREVIOUS USER MODE
1353 006412 010701      ↑112: SCOPE
1354 006414 012737 030000 177776      MOV      #KM+PUM,#PSW      ;KERNEL MODE!!! PREV USER MODE!!
1355 006422 012706 000500      MOV      #KPTR,KSP        ;SET KERNEL STACK PTR
1356 006426 012746 000700      MOV      #UPTR,-(KSP)
1357 006434 012746 001002      MTPI    USP              ;SET USER STACK PTR
1358 006440 012746 177777      MOV      #TEMP,-(KSP)     ;PUT ADDRESS ON THE STACK
1359 006444 005037 001002      MOV      #-1,-(KSP)       ;PUT DATA ON THE STAK
1360 006450 006636      CLR      #TEMP            ;PRESET DATA
1361 006452 022706 000500      MTPI    #-(KSP)+         ;MOVE #-1 TO TEMP
1362 006456 001401      CMP      #KPTR,KSP        ;CHECK STACK PTR AFTER MTPD
1363 006460 000000      BEQ      .+4
1364 006462 005267 172314      HLT
1365 006466 001401      INC      TEMP              ;ERROR! INCORRECT STACK PTR AFTER MTPD
1366 006470 000000      BEQ      .+4              ;CHECK THAT DATA WAS MOVED TO TEMP
1367 006472 006506      HLT
1368 006474 022716 000700      MFPI    USP              ;ERROR! DATA NOT IN TEMP
1369 006500 001401      CMP      #UPTR,(KSP)      ;GET USER STACK PTR
1370 006502 000000      BEQ      .+4              ;CHECK THAT USER STACK PTR NOT CHANGED
1371      HLT                    ;BY MTPD INSTRUCTION
1372      END: TST      PASCNT      ;ERROR! USP WAS CHANGED BY MTPD INST.
1373 006510 001410      BEQ      DONE
1374 006512 005267 172262      INC      ICNT              ;FIRST PASS?
1375 006516 026727 172256 000144      CMP      ICNT,#100.       ;YES, SKIP ITERATIONS THIS TIME
1376 006524 001402      BEQ      DONE              ;INCREMENT PASS COUNT
1377 006526 000167 172432      JMP      BEGIN             ;100 PASSES COMPLETED?
1378 006532 005267 172256      INC      PASCNT            ;TO ENABLE ITERATIONS ON LATER PASSES
1379 006536 032767 010000 171024      BIT      #10000,SWR        ;INHIBIT BELL AND '*'?
1380 006544 001401      BEQ      .+4
1381 006546 000422      BR       LOGICT
1382 006550 012767 000007 171010      MOV      #7,TPB           ;RING BELL
1383 006556 105767 171002      TSTB    TPS
1384 006562 100375      BPL     .-4
1385 006564 012767 000052 170774      MOV      #52,TPB         ;PRINT '*' FOR PASS INDICATION
1386 006572 105767 170766      TSTB    TPS
1387 006576 100375      BPL     .-4
1388 006600 012767 000177 170760      MOV      #177,TPB
1389 006606 105767 170752      TSTB    TPS
1390 006612 100375      BPL     .-4
1391 006614 013701 000042      LOGICT: MOV      #42,%1      ;RETURN TO MONITOR?
1392 006620 001405      BEQ      LOGICE
1393 006622 000005      RESET
1394 006624 004711      SENDAD: JSR      7,(1)      ;RETURN!
1395 006626 000240      NOP
1396 006630 000240      NOP
1397 006632 000240      NOP
1398 006634 005000      LOGICE: CLR      R0        ;DELAY FOR ACT11
1399 006636 005200      INC      R0
    
```

C03

DBKTD-C KT11-D PROCESSORS STATES TEST MACY11 27(1006) 02-FEB-77 10:09 PAGE 28
DBKTD C.P11 02-FEB-77 09:11

1400 006640 001376
1401 006642 000167 172150
1402 000001

BNE -2
JMP START
.END

BEGIN = 001164	REDPTR= 000736	T110A 006246	T26 002662	T56 004640
BIT13 = 020000	REG = 000000	T110AA 006240	T30 002752	T56A 004714
BIT14 = 040000	RESVEC= 000010	T111 006306	T31 003046	T56AA 004712
BIT15 = 100000	SCOPE = 010701	T111A 006350	T31C 003072	T57 004772
BIT6 = 000100	SLR = 177774	T111AA 006342	T32A 003144	T57A 005076
C = 000001	START 001016	T112 006412	T35 003214	T57AA 005026
DONE 006532	SWR = 177570	T12 001504	T36 003254	T57EX 005102
EMTVEC= 000030	TBIT = 000020	T12A 001542	T36A 003312	T60 005104
END 006504	TBITVE= 000014	T12AA 001534	T36AA 003310	T62 005132
ERRVEC= 000004	TEMP 001002	T13 001574	T40 003334	T65 005172
FTITLE 001012	TITLE 001076	T13A 001626	T40A 003404	T66 005262
HLT = 000000	TKB = 177562	T13AA 001620	T40AA 003376	T7 001374
ICNT 001000	TKS = 177560	T15 001772	T41 003460	T7A 001434
IM = 100000	TPB = 177566	T15A 002052	T41A 003526	T7AA 001426
IM1 = 040000	TPS = 177564	T15AA 002044	T41B 003546	T70 005310
IOIVEC= 000020	TPVEC = 000064	T17 002134	T41BB 003576	T73 005350
KM = 000000	TRAPVE= 000034	T17A 002170	T43 003624	T74 005440
KPTR = 000500	TRT = 000003	T17B 002202	T43A 003676	T75 005470
KSP = %000006	TRTVEC= 000014	T17C 002222	T44 003744	T76 005520
LOGICE 006634	T1 001170	T17D 002260	T45 004020	UM = 140000
LOGICT 006614	T102 005632	T17E 002302	T5 001222	UPTR = 000700
N = 000010	T103 005716	T17ERR 002402	T5A 001266	USP = %000006
PASCNT 001014	T104 006014	T17F 002322	T5AA 001260	V = 000002
PKM = 000000	T104A 006042	T17G 002354	T50 004104	YELPTR= 001000
PRTY3 = 000140	T106 006052	T17X 002410	T51 004204	Z = 000004
PRTY4 = 000200	T106A 006104	T18 001666	T52 004302	\$ENDAD 006624
PRTY7 = 000340	T107 006124	T19 001730	T54 004404	= 006646
PSW = 177776	T107A 006154	T21 002426	T55 004522	
PUM = 030000	T107AA 006132	T22 002510	T55A 004576	
PWRUP 001154	T110 006204	T25 002572	T55AA 004574	

. ABS. 006646 000

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0

MULE: DBKTDC, MULE: DBKTDC/SOL=DSKZ:SYSMAC.SML, MULE: DBKTDC.P11
 RUN-TIME: 7 8 .1 SECONDS
 RUN-TIME RATIO: 251/15=16.0
 CORE USED: 31K (61 PAGES)