

# KT11-D

ACCESS KEYS TEST  
MD-11-DBKTB-B

EP-DBKTB-B-DL-B  
COPYRIGHT © 1977  
FICHE 1 OF 1

APR 1977  
**digital**  
MADE IN USA

This microfiche strip contains 15 frames of data. Each frame displays a grid of information, likely test results or system status, with columns and rows of text and numbers. The frames are arranged vertically from top to bottom.

100  
100



801

DB\*TB-B KT11-D ACCESS KEYS TEST  
DBKT88.P11 02-FEB-77 09:09

MACY11 27(1006) 02-FEB-77 10:04 PAGE 2

.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DBKTB-B-D  
PRODUCT NAME: KT11-D ACCESS KEYS 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 PROGRAM CHECKS THE OPERATION OF EACH ACCESS KEY FOR EACH OF THE FOUR UNIBUS CYCLES (OR COMBINATION OF CYCLES) WHICH MAY REFERENCE AN ADDRESS THRU SEGMENTATION. THESE CYCLES ARE DATI, DATO (NO DATIP), DATIP-DATO, AND DATIP-DATOB. EACH OF THESE CASES IS TESTED WITH AND WITHOUT MEMORY MANAGEMENT ENABLE SET. THUS EIGHT CASES ARE TESTED FOR EACH KEY. SR0, SR1, SR2, THE CORRESPONDING PDR'S, AND THE PROPER EXECUTION OR PREVENTION OF EXECUTION OF THE INSTRUCTION ARE CHECKED IN EACH CASE.

## 2.0 REQUIREMENTS

## 2.1 EQUIPMENT

PDP 11/40 WITH KT11-D OPTION

## 2.2 STORAGE

THE PROGRAM REQUIRES 5K OF MEMORY, STARTING AT LOCATION 0.

## 3.0 LOADING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABS LOADER.

## 4.0 STARTING PROCEDURE

## 4.1 NORMAL DIAGNOSTIC OPERATION

LOAD ADDRESS 200.  
SET DESIRED SWITCH REGISTER SETTINGS (ALL DOWN FOR WORST CASE).  
PRESS START.  
THE PROGRAM WILL RING THE BELL AND PRINT AN '\*' ON COMPLETION OF A PASS.

## 4.2 SINGLE SUBTEST LOOP (TESTX)

LOAD ADDRESS 210.  
PRESS START.  
AT THE FIRST HALT, LOAD THE ADDRESS OF THE DESIRED SUBTEST (THE ADDRESS OF THE TESTXX TAG) INTO THE SWITCH REGISTER.  
THEN PRESS "CONTINUE".  
AT THE SECOND HALT, SET THE OPERATIONAL SWITCH SETTINGS DESIRED (SW11 MUST BE SET TO ZERO). THEN PRESS CONTINUE.

## 5.0 OPERATING PROCEDURE

## 5.1 OPERATIONAL SWITCH SETTINGS

SW15=1 OR UP-- HALT ON ERROR  
SW14=1 OR UP-- SCOPE LOOP  
SW13=1 OR UP-- INHIBIT PRINTOUT  
SW11=1 OR UP-- INHIBIT ITERATIONS  
SW10=1 OR UP-- HALT AT END OF CURRENT TEST  
NEXT TEST NUMBER IN DATA LIGHTS

## 5.2 SUBROUTINE ABSTRACTS

## 5.2.1 SCOPE

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

## 5.2.2 HLT

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

## 5.2.3 TRAPCATCHER

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

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 COUNTER WHEN THE TRAP OR INTERRUPT OCCURRED.

## 5.2.4 TESTX (SINGLE SUBTEST LOOP)

THIS ROUTINE ALLOWS A SINGLE SUBTEST TO BE RUN CONTINUOUSLY FOR SCOPE LOOP PURPOSES. WHILE A SCOPE LOOP SWITCH OPTION EXISTS, IT REQUIRES THAT YOU ARE WITHIN THE TEST IN WHICH YOU WISH TO LOOP. IN SOME CASES (SUCH AS WITH INTERMITTENT FAILURES) THAT'S NOT EASY TO DO. THIS SUBROUTINE ALLOWS YOU TO LOAD THE ADDRESS OF ANY SUBTEST AT THE HALT AND THEN GO DIRECTLY TO THAT TEST.

## 5.2.5 ENTSRV (EMT DECODER)

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

## 5.2.6 CLRALL

THIS ROUTINE CLEARS ALL THE PAR'S AND PDR'S OF THE KT11-D, AS WELL AS SRD.

## 5.2.7 RWALL

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

## 5.2.8 SETUP

THIS ROUTINE FIRST CALLS RWALL TO MAP ALL THE PAGES 4K, RW, BANK 0. IT THEN SETS THE KEY FOR KERNEL PAGE 1 TO WHATEVER VALUE WAS STORED ON THE STACK BEFORE THE ROUTINE WAS CALLED. THIS ALLOWS A REFERENCE TO PAGE 1 TO TEST THE DESIRED ACCESS KEY. FINALLY, KERNEL PAGE 7 IS MAPPED TO THE EXTERNAL BANK.

## 5.3 PROGRAM AND/OR OPERATOR ACTION

## 5.3.1 SA 200 (NORMAL DIAGNOSTIC OPERATION)

THE PROGRAM EXECUTES SEVERAL TESTS OF EACH KEY. TESTS 5 THRU 10 ARE CYCLED THRU 3 TIMES, ONCE FOR EACH OF THE KEYS WHICH GIVES A NON-RESIDENT ABORT. AT THE END OF EACH PASS THRU THE DIAGNOSTIC THE BELL IS RUNG, AND AN '\*' IS PRINTED.

## 5.3.2 SA 210 (SINGLE SUBTEST LOOP)

THIS STARTING ADDRESS ALLOWS THE USER TO RUN A SINGLE SUBTEST REPEATEDLY BY GIVING THE ADDRESS OF THE DESIRED SUBTEST AT THE FIRST HALT. IF SW11 IS SET TO A ONE, NORMAL TEST EXECUTION WILL BE RESUMED.

## 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.

## 7.0 RESTRICTIONS

PROGRAM MUST BE LOADED INTO LOWER 5K OF MEMORY.

## 8.0 MISCELLANEOUS

## 8.1 EXECUTION TIME

EACH PASS TAKES APPROXIMATELY 1 MINUTE WITH CORE MEMORY.

## 9.0 PROGRAM DESCRIPTION

THE PROGRAM RUNS SEVERAL SEPARATE TESTS OF EACH ACCESS KEY. DATI, DATO (NO DATIP), DATIP-DATO, AND DATIP-DATOB ARE CHECKED FOR EACH KEY, WITH AND WITHOUT MEMORY MANAGEMENT ENABLE SET. THE BELL IS RUNG AND '\*' PRINTED AT THE END OF EACH PASS.

%

;COPYRIGHT 1972,1977 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754  
;TEST OF THE KT11-D ACCESS KEYS

;OPERATING INSTRUCTIONS

1. LOAD TEST USING THE ABSOLUTE LOADER
2. LOAD SA 200
3. SET SR TO INITIAL SETTINGS
4. PRESS START

;DYNAMIC SWITCH REGISTER SETTINGS ARE:

;SW15=1 CAUSES HALT ON ERROR  
 ;SW14=1 CAUSES SCOPE LOOPING  
 ;SW13=1 INHIBITS ERROR PRINTOUT  
 ;SW11=1 INHIBITS ITERATIONS  
 ;SW10=1 HALT AT END OF CURRENT TEST WITH NEXT TEST NUMBER  
 ; IN DATA LIGHTS. PRESS CONTINUE TO ADVANCE TO NEXT TEST.

;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  
 SR=177570  
 PS=177776  
 STATUS=PS  
 HLT=104006

;LOAD TRAP CATCHER IN LOCATIONS 0 THRU 377  
 ;EACH VECTOR ADDRESS IS LOADED WITH THE ADDRESS  
 ;OF THE NEXT LOCATION, AND THE NEXT LOCATION IS LOADED  
 ;WITH A HALT INSTRUCTION (000000)

;LOAD VECTOR AREA

. =30  
 EMTSRV  
 340  
 . =34  
 SCOPEC  
 0

;LOAD ACT11 HOOKS

. =46  
 \$ENDAD  
 . =52  
 000000

;LOAD STARTING AREA

. =200  
 JMP START

225  
 226  
 227  
 228  
 229  
 230  
 231  
 232  
 233  
 234  
 235  
 236  
 237  
 238  
 239  
 240  
 241  
 242  
 243 104400  
 244 000240  
 245 000000  
 246 000001  
 247 000002  
 248 000003  
 249 000004  
 250 000005  
 251 000006  
 252 000007  
 253 000006  
 254 000007  
 255 177570  
 256 177776  
 257 177776  
 258 104006  
 259  
 260  
 261  
 262  
 263  
 264  
 265  
 266 000030  
 267 000030 006402  
 268 000032 000340  
 269 000034 000034  
 270 000034 005670  
 271 000036 000000  
 272  
 273 000046  
 274 000046 005400  
 275 000052 000052  
 276 000052 000000  
 277  
 278  
 279 000200  
 280 000200 000167 001746

H01

```

281      000210 000210           . = 210
282      000210 000167 005356    JMP      TESTX
283
284      ;LOAD DATA AREA
285      001000 001000           . = 1000
286      001000 000000    KSTACK: 0
287      002000 002000           . = +776
288      002000 000000    USTACK: 0
289      002002 000000 000000 000000 .WORD 0,0,0,0
290      002010 000000
291      002012 177564    TCSR:    177564
292      002014 177566    TDBR:    177566
293      002016 177572    SR0:    177572
294      002020 177574    SR1:    177574
295      002022 177576    SR2:    177576
296      002024 000250    KTVEC:  250
297      002026 000252    KTSTA:  252
298      002030
299      002030 177600    ADRTAB:
300      002032 177602    UPDR0:  177600
301      002034 177604    UPDR1:  177602
302      002036 177606    UPDR2:  177604
303      002040 177610    UPDR3:  177606
304      002042 177612    UPDR4:  177610
305      002044 177614    UPDR5:  177612
306      002046 177616    UPDR6:  177614
307      002050 177618    UPDR7:  177616
308      002052 177642    UPAR0:  177640
309      002054 177644    UPAR1:  177642
310      002056 177646    UPAR2:  177644
311      002060 177648    UPAR3:  177646
312      002062 177652    UPAR4:  177650
313      002064 177654    UPAR5:  177652
314      002066 177656    UPAR6:  177654
315      002070 172300    UPAR7:  177656
316      002072 172302    KPDR0:  172300
317      002074 172304    KPDR1:  172302
318      002076 172306    KPDR2:  172304
319      002100 172310    KPDR3:  172306
320      002102 172312    KPDR4:  172310
321      002104 172314    KPDR5:  172312
322      002106 172316    KPDR6:  172314
323      002110 172318    KPDR7:  172316
324      002112 172342    KPAR0:  172340
325      002114 172344    KPAR1:  172342
326      002116 172346    KPAR2:  172344
327      002120 172350    KPAR3:  172346
328      002122 172352    KPAR4:  172350
329      002124 172354    KPAR5:  172352
330      002126 172356    KPAR6:  172354
331      002126 002126    KPAR7:  172356
332      002130 000000    ADRENO= -2
333      002132 000000    PASCNT: 0
334      002134 177573    FTITLE: 0
335      002136 177575    SR0H:   177573
336      002140 177577    SR1H:   177575
336      002140 177577    SR2H:   177577

```

;TELETYPE PRINTER CSR

;KT11-D STATUS REGISTER ADDRESSES

;KT11-D INTERRUPT VECTOR

;USER PAGE DESCRIPTOR REGISTER ADDRESSES

;USER PAGE ADDRESS REGISTER ADDRESSES

;KERNEL PAGE DESCRIPTOR REGISTER ADDRESSES

;KERNEL PAGE ADDRESS REGISTER ADDRESSES

;TITLE PRINTED FLAG

;KT11-D STATUS REGISTER HIGH BYTE ADDRESSES



```

337 002142 000000
338 002144 000000 000004
339 002150 125252
340
341
342
343
344 002152 005037 177776
345 002156 012706 001000
346 002162 012737 140000 177776
347 002170 012706 002000
348 002174 005037 177776
349 002200 012767 002000 003562
350 002206 012767 002272 003560
351 002214 005067 177722
352 002220 012767 000001 004474
353 002226 023737 000042 000046
354 002234 001416
355 002236 005767 177670
356 002242 001013
357 002244 004767 004240
358 002250 004767 004302
359 002254 005414
360 002256 004767 004226
361 002262 005267 177644
362 002266 000401

```

```

NRCNT: 0
NRKEYS: 0,4
DESTAD: 125252

```

```

;COUNTER FOR TEST OF THE 3 NR KEYS
;VALUES OF THE 3 NON RESIDENT KEYS
;LOCATION USED FOR READS AND WRITES TO CHECK
;EXECUTION OR ABORTING AT CORRECT POINT

```

```

;SET UP FOR START OF TESTS
START:

```

```

CLR 2#PS
MOV #KSTACK,SP
MOV #140000,2#PS
MOV #USTACK,SP
CLR 2#PS
MOV #2000,ICOUNT
MOV #TEST1+2,RETURN
CLR NRCNT
MOV #1,TESTCT
CMP 2#42,2#46
BEQ TEST1+2
TST FTITLE
BNE TEST1+2
JSR PC,CRLF
JSR PC,TYPE
MTIT
JSR PC,CRLF
INC FTITLE
BR .+4

```

```

;SETUP KERNEL STACK
;SETUP USER STACK POINTER
;INITIALIZE ITERATION COUNT
;SETUP SCOPE AND ITERATION LOOP RETURN
;INITIALIZE FOR NR TEST
;SET UP TEST SEQUENCE
;ARE WE IN ACT11 AUTOMATIC MODE?
;YES, SKIP TITLE
;TITLE PRINTED
;YES, SKIP
;PRINT TITLE

```

```

363
364
365
366
367 002270 104400
368 002272 012706 001000
369 002276 005077 177514
370 002302 004767 004330
371 002306 000001
372 002310 104006
373 002312 012746 000002
374 002316 004767 003220
375
376
377 002322 005726
378 002324 012777 002440 177472
379 002332 005077 177470
380 002336 012767 125252 177604
381 002344 012701 022150
382
383 002350 005277 177442
384 002354 022721 125252
385 002360 001404
386 002362 005377 177430
387 002366 104006
388 002370 000427
389 002372 017702 177420
390 002376 105377 177414
391 002402 022702 000017
392 002406 001401
393 002410 104006
394
395
396 002412 022777 002412 177402
397 002420 001401
398 002422 104006
399
400 002424 022777 077402 177440
401 002432 001401
402 002434 104006
403
404 002436 000404
405 002440 042777 000001 177350
406 002446 104006
407
408 002450 016777 177352 177346
409 002456 005077 177344
410 002462 005077 177330
411 002466 005037 177776
412
413
414
415
416 002472 104400
417 002474 012706 001000
418 002500 005077 177312
    
```

```

;SHOW THAT DATI TO A RRO PAGE (ACF=2) NEITHER TRAPS NOR ABORTS
;SHOW THAT THE KT11-D STATUS REGISTERS CONTINUE TO TRACK, AND THAT
;THE PDR CORRESPONDING TO THE REFERENCE IS CORRECT
TEST1: SCOPE
        MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
        CLR     @SRO            ;INITIALIZE SRO
        JSR     PC,ORDER        ;CHECK TEST SEQUENCE
        I       1              ;TEST NUMBER
        HLT     @SRO            ;TEST EXECUTED OUT OF SEQUENCE
        MOV     #2,-(SP)        ;PUSH RRO KEY ON STACK
        JSR     %7,SETUP        ;MAKE KERNEL PAGE 1 RRO, BANK 0
        ;MAKE KERNEL PAGE 7 RW, EXTERNAL
        ;MAKE ALL OTHER PAGES RW, BANK 0
        TST     (SP)+           ;RESTORE STACK
        MOV     @RET1,@KTVEC    ;SETUP ABORT RETURN IN CASE
        CLR     @KTSTA
        MOV     #125252,@DESTAD ;SETUP LOCATION TO BE REFERENCED
        MOV     @DESTAD+20000,R1 ;R1 CONTAINS VIRTUAL ADDRESS OF LOCATION TO
        ;BE REFERENCED THRU KERNEL PAGE 1
        INC     @SRO            ;TURN ON KT11-D
        CMP     #125252,(R1)+    ;DATI TO RRO PAGE
        BEQ     @CMPOK1         ;BRANCH IF CORRECT VALUE WAS READ
        DEC     @SRO            ;ON ERROR, TURN OFF KT11-D
        HLT     @SRO            ;RELOCATION FAILED THRU KERNEL PAGE 1
        BR      @DONE1
CMPOK1: MOV     @SRO,R2         ;SAVE CONTENTS OF SRO
        DECB   @SRO            ;TURN OFF KT11-D
        CMP     #17,R2          ;CHECK SAVED CONTENTS OF SRO
        BEQ     .+4
        HLT     @SRO            ;SRO INCORRECT-SHOULD HAVE
        ;TRACKED REFERENCE TO PAGE 0,
        ;WHICH GOT THE ADDRESS OF SRO
        ;CHECK SR2
        CMP     #,@SR2
        BEQ     .+4
        HLT     @SRO            ;SR2 INCORRECT-SHOULD TRACK EVEN
        ;WHEN KT11-D IS OFF
        CMP     #77402,@KPDR1   ;CHECK PDR FOR
        BEQ     .+4             ;THE RRO PAGE REFERENCED
        HLT     @SRO            ;KPDR1 INCORRECT-SHOULD NOT
        ;HAVE BEEN CHANGED
        BR      @DONE1
RET1:   BIC     #1,@SRO        ;TURN OFF KT11-D
        HLT     @SRO            ;DATI TO RRC PAGE CAUSED
        ;A TRAP OR ABORT
DONE1:  MOV     @KTSTA,@KTVEC   ;RESTORE TRAP RETURN TO CAUSE HALT
        CLR     @KTSTA         ;ON AN UNEXPECTED TRAP
        CLR     @SRO          ;INITIALIZE SRO
        CLR     @#PS          ;INITIALIZE PROCESSOR STATUS

;SHOW THAT A DATO (NO DATIP) TO A RRO PAGE (ACF=2) ABORTS
;SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT THE PDR
;CORRESPONDING TO THE REFERENCE IS CORRECT
TEST2: SCOPE
        MOV     #KSTACK,SP      ;INITIALIZE KERNEL STACK POINTER
        CLR     @SRO            ;INITIALIZE SRO
    
```

419	002504	004767	004126		JSR	PC, ORDER		:CHECK TEST SEQUENCE
420	002510	000002			2			:TEST NUMBER
421	002512	104006			HLT			:TEST EXECUTED OUT OF SEQUENCE
422	002514	012746	000002		MOV	#2, -(SP)		:PUSH RRO KEY ON STACK
423	002520	004767	003016		JSR	%7, SETUP		:MAKE KERNEL PAGE 1 RRO, BANK 0
424								:MAKE KERNEL PAGE 7 RW, EXTERNAL
425								:MAKE ALL OTHER PAGES RW, BANK 0
426	002524	005726			TST	(SP)+		:RESTORE STACK POINTER
427	002526	012777	002572	177270	MOV	#RET4, @KTVEC		:SETUP ABORT RETURN
428	002534	005077	177266		CLR	@KTSTA		
429	002540	005067	177404		CLR	DESTAD		:INITIALIZE LOCATION TO BE ADDRESSED
430								:BY DATO TO RRO PAGE
431	002544	012702	022150		MOV	#DESTAD+20000, R2		:R2 CONTAINS ADDRESS OF LOCATION
432								:TO BE REFERENCED THRU KERNEL PAGE 1
433	002550	012777	000001	177240	MOV	#1, @SRO		:TURN ON KT11-D
434	002556	012722	125252		MOV	#125252, (R2)+		:DATO TO RRO PAGE-SHOULD ABORT
435	002562	005377	177230		DEC	@SRO		:TURN OFF KT11-D
436	002566	104006			HLT			:DATO TO RRO PAGE FAILED TO ABORT
437	002570	000426			BR	DONE4		
438	002572	017701	177220		MOV	@SRO, R1		:SAVE CONTENTS OF SRO
439	002576	005377	177214		DEC	@SRO		:TURN OFF KT11-D
440	002602	022701	020003		CMP	#20003, R1		:CHECK SAVED CONTENTS OF SRO
441	002606	001401			BEQ	.+4		
442	002610	104006			HLT			:SRO INCORRECT-SHOULD HAVE LOCKED
443								:ON DATO TO KERNEL PAGE 1(RRO)
444								:AND ACCESS FAULT SHOULD BE SET
445	002612	022777	002556	177202	CMP	#AD4, @SR2		:CHECK SR2
446	002620	001401			BEQ	.+4		
447	002622	104006			HLT			:SR2 INCORRECT-SHOULD HAVE LOCKED
448								:ON THE ABORTED REFERENCE, WITH THE
449								:VIRTUAL ADDRESS OF THE INSTRUCTION
450	002624	022777	077402	177240	CMP	#77402, @KPDR1		:CHECK INSTRUCTION SPACE PDR
451	002632	001401			BEQ	.+4		
452	002634	104006			HLT			:KPDR1 INCORRECT-SHOULD NOT
453								:HAVE BEEN CHANGED SINCE THE
454								:DATO DIDN'T WRITE
455	002636	005767	177306		TST	DESTAD		:MAKE CERTAIN THAT DESTINATION
456	002642	001401			BEQ	.+4		:LOCATION WAS NOT WRITTEN
457	002644	104006			HLT			:DATO TO RRO PAGE WROTE
458								:INTO THE DESTINATION LOCATION
459	002646	016777	177154	177150	MOV	KTSTA, @KTVEC		:CHANGE KT11-D TRAP RETURN
460	002654	005077	177146		CLR	@KTSTA		:TO CAUSE A HALT ON AN UNEXPECTED TRAP
461	002660	005077	177132		CLR	@SRO		
462	002664	005037	177776		CLR	@#PS		
463								
464								
465								
466								
467	002670	104400						
468	002672	012706	001000		MOV	#KSTACK, SP		:INITIALIZE KERNEL STACK POINTER
469	002676	005077	177114		CLR	@SRO		:INITIALIZE SRO
470	002702	004767	003730		JSR	PC, ORDER		:CHECK TEST SEQUENCE
471	002706	000003			3			:TEST NUMBER
472	002710	104006			HLT			:TEST EXECUTED OUT OF SEQUENCE
473	002712	012746	000002		MOV	#2, -(SP)		:PUSH RRO KEY ON STACK
474	002716	004767	002620		JSR	%7, SETUP		:MAKE KERNEL PAGE 1 RROT, BANK 0

:SHOW THAT A DATIP, DATO SEQUENCE TO A RRO PAGE (ACF=2) ABORTS  
 :SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT THE PDR  
 :CORRESPONDING TO THE REFERENCE IS CORRECT  
 TEST3: SCOPE





```

531 003136 005067 177006          CLR  DESTAD          ; INITIALIZE LOCATION TO BE ADDRESSED
532                                ; BY DATIP, DATOB TO RRO PAGE
533 003142 012704 022150          MOV  #DESTAD+20000,R4 ; R4 CONTAINS VIRTUAL ADDRESS OF LOCATION
534                                ; TO BE REFERENCED THRU KERNEL PAGE 1
535 003146 052777 000001 176642  AD6:  BIS  #1,SR0 ; TURN ON KT11-D
536 003154 105224                INCB  (R4)+          ; DATIP, DATOB TO RROT PAGE
537 003156 005377 176634          DEC  SR0           ; TURN OFF KT11-D
538 003162 104006                HLT                    ; DATIP, DATO TO RROT PAGE FAILED TO ABORT
539 003164 000426                BR    DONE6
540 003166 017701 176624          RET6: MOV  SR0,R1    ; SAVE CONTENTS OF SRO
541 003172 005377 176620          DEC  SR0           ; TURN OFF KT11-D
542 003176 022701 020003          CMP  #20003,R1     ; CHECK SAVED CONTENTS OF SRO
543 003202 001401                BEQ                    ;
544 003204 104006                HLT                    ; SRO INCORRECT-SHOULD HAVE LOCKED ON
545                                ; DATOB TO KERNEL PAGE 1 (RRO)
546                                ; ACCESS FAULT SHOULD BE SET
547 003206 022777 003154 176606  CMP  #AD6,SR2      ; CHECK SR2
548 003214 001401                BEQ                    ;
549 003216 104006                HLT                    ; SR2 INCORRECT-SHOULD HAVE LOCKED
550                                ; ON THE ABORTED REFERENCE, WITH THE
551                                ; VIRTUAL ADDRESS OF THE INSTRUCTION
552 003220 022777 077402 176644  CMP  #77402,PKPDR1 ; CHECK PDR
553 003226 001401                BEQ                    ;
554 003230 104006                HLT                    ; KPDR1 INCORRECT - SHOULD NOT HAVE
555                                ; BEEN CHANGED-DATIP IS ABORTED
556                                ; SINCE IT MUST BE FOLLOWED BY A DATO
557 003232 005767 176712          TST  DESTAD        ; MAKE CERTAIN THAT DESTINATION
558 003236 001401                BEQ                    ; LOCATION WAS NOT WRITTEN
559 003240 104006                HLT                    ; DATOB TO RRO PAGE WROTE INTO
560                                ; THE DESTINATION LOCATION
561 003242 016777 176560 176554  DONE6: MOV  KTSTA,KTVEC ; CHANGE KT11-D FAULT
562 003250 005077 176552          CLR  #KTSTA        ; RETURN TO CAUSE A HALT ON AN
563 003254 005077 176536          CLR  SR0           ; UNEXPECTED TRAP
564 003260 005037 177776          CLR  #PS
565
566                                ; THE FOLLOWING TESTS (5-10) ARE RUN FOR BOTH OF THE NON-RESIDENT
567                                ; KEYS - A PASS IS MADE FOR KEY 0, THEN A PASS IS MADE FOR KEY 4,
568                                ; THE CURRENT KEY IS STORED ON THE STACK
569                                ; SHOW THAT DATI TO A NR PAGE ABORTS WITHOUT COMPLETING
570                                ; SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT
571                                ; THE PDR CORRESPONDING TO THE REFERENCE IS CORRECT
572                                TESTS: SCOPE
573 003264 104400                MOV  #KSTACK,SP    ; INITIALIZE KERNEL STACK POINTER
574 003266 012706 001000          CLR  SR0           ; INITIALIZE SRO
575 003272 005077 176520          JSR  PC,ORDER      ; CHECK TEST SEQUENCE
576 003276 004767 003334          S    5             ; TEST NUMBER
577 003302 000005                HLT                    ; TEST EXECUTED OUT OF SEQUENCE
578 003304 104006                CLR  #KSTACK        ; PUT 0 ON STACK AS FIRST NR KEY TO BE TESTED
579 003306 005037 001000          CLR                    ; THIS INSTRUCTION IS SKIPPED WHEN TESTING THE
580                                ; OTHER WHICH IS SETUP AFTER TEST30
581 003312 012706 001000          RERUNA: MOV #KSTACK,SP
582 003316 005077 176474          CLR  SR0
583 003322 004767 002214          JSR  %7,SETUP      ; MAKE KERNEL PAGE 1 NR, BANK 0
584                                ; MAKE KERNEL PAGE 7 RW, EXTERNAL
585                                ; MAKE ALL OTHER PAGES RW, BANK 0
586 003326 012777 003372 176470  MOV  #RET21,KTVEC  ; SETUP ABORT RETURN

```

587	003334	005077	176466		CLR	2KTSTA		
588	003340	005003			CLR	R3		; INITIALIZE DESTINATION LOCATION
589	003342	012757	125252	176600	MOV	2125252,DESTAD		; INITIALIZE SOURCE LOCATION
590	003350	012701	022150		MOV	2DESTAD+20000,R1		; R1 CONTAINS VIRTUAL ADDRESS OF LOCATION
591								; TO BE REFERENCED THRU KERNEL PAGE 1
592	003354	005277	176436		INC	2SR0		; TURN ON KT11-D
593	003360	012103			MOV	(R1)+,R3		; DATI TO NR PAGE - SHOULD ABORT
594	003362	005377	176430		DEC	2SR0		; ON ERROR, TURN OFF KT11-D
595	003366	104006			HLT			; NO ABORT ON DATI TO A NON-RESIDENT PAGE
596	003370	000430			BR	DONE21		
597	003372	017702	176420		MOV	2SR0,R2		; SAVE CONTENTS OF SR0
598	003376	105377	176414		DECB	2SR0		; TURN OFF KT11-D
599	003402	022702	100003		CMP	2100003,R2		; CHECK SAVED CONTENTS OF SR0
600	003406	001401			BEQ	.+4		
601	003410	104006			HLT			; SR0 INCORRECT-SHOULD HAVE
602								; LOCKED ON REFERENCE TO
603								; KERNEL PAGE 1 WHICH WAS NON-RESIDENT
604	003412	022777	003360	176402	CMP	2AD21,2SR2		; CHECK SR2
605	003420	001401			BEQ	.+4		
606	003422	104006			HLT			; SR2 INCORRECT-SHOULD HAVE LOCKED ON
607								; NR REFERENCE
608	003424	017705	176442		MOV	2KPDR1,R5		; MOVE CONTENTS OF KPDR1 TO R5
609	003430	042705	000007		BIC	27,R5		; TO MASK OFF ACCESS KEY
610	003434	022705	077400		CMP	277400,R5		; CHECK PDR FOR
611	003440	001401			BEQ	.+4		; THE NR PAGE REFERENCED (BITS 0-2 MASKED OUT)
612	003442	104006			HLT			; KPDR1 INCORRECT-SHOULD NOT
613								; HAVE BEEN CHANGED
614	003444	005703			TST	R3		; CHECK DESTINATION LOCATION TO SEE
615	003446	001401			BEQ	.+4		; IF INSTRUCTION ALTERED IT BEFORE ABORTING
616	003450	104006			HLT			; INSTRUCTION COMPLETED BEFORE ABORT OCCURRED
617	003452	016777	176350	176344	MOV	KTSTA,2KTVEC		; RESTORE TRAP RETURN TO CAUSE HALT
618	003460	005077	176342		CLR	2KTSTA		; ON AN UNEXPECTED TRAP
619	003464	005077	176326		CLR	2SR0		; INITIALIZE SR0
620	003470	005037	177776		CLR	2#PS		; INITIALIZE PROCESSOR STATUS
621								
622								; SHOW THAT A DATO (NO DATIP) TO A NR PAGE
623								; ABORTS WITHOUT COMPLETING THE DATO
624								; SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT THE PDR
625								; CORRESPONDING TO THE REFERENCE IS CORRECT
626	003474	104400			TEST6:	SCOPE		
627	003476	012706	001000		MOV	2KSTACK,SP		; INITIALIZE KERNEL STACK POINTER
628	003502	005077	176310		CLR	2SR0		; INITIALIZE SR0
629	003506	004767	003124		JSR	PC,ORDER		; CHECK TEST SEQUENCE
630	003512	000006			6			; TEST NUMBER
631	003514	104006			HLT			; TEST EXECUTED OUT OF SEQUENCE
632	003516	004767	002020		JSR	27,SETUP		; MAKE KERNEL PAGE 1 NR, BANK 0
633								; MAKE KERNEL PAGE 7 RW, EXTERNAL
634								; MAKE ALL OTHER PAGES RW, BANK 0
635	003522	012777	003570	176274	MOV	2RET23,2KTVEC		; SETUP ABORT RETURN
636	003530	005077	176272		CLR	2KTSTA		
637	003534	005067	176410		CLR	DESTAD		; INITIALIZE LOCATION TO BE ADDRESSED
638								; BY DATO TO NR PAGE
639	003540	012701	022150		MOV	2DESTAD+20000,R1		; R1 CONTAINS ADDRESS OF LOCATION
640								; TO BE REFERENCED THRU KERNEL PAGE 1
641	003544	112777	000001	176244	MOV	21,2SR0		; TURN ON KT11-D
642	003552	012721	125252		MOV	2125252,(R1)+		; DATO TO NR PAGE-SHOULD ABORT

643	003556	042777	000001	176232		BIC	#1,SR0	:TURN OFF KT11-D
644	003564	104006				HLT		:DATO TO NR PAGE FAILED TO ABORT
645	003566	000431				BR	DONE23	
646	003570	017702	176222		RET23:	MOV	SR0,R2	:SAVE CONTENTS OF SRO
647	003574	005377	176216			DEC	SR0	:TURN OFF KT11-D
648	003600	022702	100003			CMP	#100003,R2	:CHECK SAVED CONTENTS OF SRO
649	003604	001401				BEQ	.+4	
650	003606	104006				HLT		:SRO INCORRECT-SHOULD HAVE LOCKED
651								:ON DATO TO KERNEL PAGE 1(NR)
652								:NR FAULT SHOULD BE SET
653	003610	022777	003552	176204		CMP	#AD23,SR2	:CHECK SR2
654	003616	001401				BEQ	.+4	
655	003620	104006				HLT		:SR2 INCORRECT-SHOULD HAVE LOCKED
656								:ON THE ABORTED REFERENCE, CONTAINING THE
657								:VIRTUAL ADDRESS OF THE INSTRUCTION
658	003622	017703	176244			MOV	AKPDR1,R3	:MOVE CONTENTS OF KPDR1 TO R3
659	003626	042703	000007			BIC	#7,R3	:TO MASK OFF THE ACCESS KEY
660	003632	022703	077400			CMP	#77400,R3	:CHECK PDR
661	003636	001401				BEQ	.+4	: (BITS 0-2 MASKED OUT)
662	003640	104006				HLT		:KPDR1 INCORRECT-SHOULD NOT HAVE
663								:BEEN CHANGED
664	003642	005767	176302			TST	DESTAD	:MAKE CERTAIN THAT DESTINATION
665	003646	001401				BEQ	.+4	:LOCATION WAS NOT WRITTEN
666	003650	104006				HLT		:DATO TO NR PAGE WROTE
667								:INTO THE DESTINATION LOCATION
668	003652	016777	176150	176144	DONE23:	MOV	KTSTA,AKTVEC	:CHANGE KT11-D FAULT RETURN
669	003660	005077	176142			CLR	AKTSTA	:TO CAUSE A HALT ON AN UNEXPECTED TRAP
670	003664	005077	176126			CLR	SR0	
671	003670	005037	177776			CLR	PS	
672								
673								:SHOW THAT A DATIP, DATO SEQUENCE TO A NR PAGE WORD ABORTS
674								:SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT THE PDR
675								:CORRESPONDING TO THE REFERENCE IS CORRECT
676	003674	104400			TEST7:	SCOPE		
677	003676	012706	001000			MOV	#KSTACK,SP	:INITIALIZE KERNEL STACK POINTER
678	003702	005077	176110			CLR	SR0	:INITIALIZE SRO
679	003706	004767	002724			JSR	PC,ORDER	:CHECK TEST SEQUENCE
680	003712	000007				7		:TEST NUMBER
681	003714	104006				HLT		:TEST EXECUTED OUT OF SEQUENCE
682	003716	004767	001620			JSR	7,SETUP	:MAKE KERNEL PAGE 1 NR, BANK 0
683								:MAKE KERNEL PAGE 7 RW, EXTERNAL
684								:MAKE ALL OTHER PAGES RW, BANK 0
685	003722	012777	003766	176074		MOV	#RET25,AKTVEC	:SETUP ABORT RETURN
686	003730	005077	176072			CLR	AKTSTA	
687	003734	005067	176210			CLR	DESTAD	:INITIALIZE LOCATION TO BE ADDRESSED
688								:BY DATIP, DATO TO NR PAGE
689	003740	012703	022152			MOV	#DESTAD+20002,R3	:R3 CONTAINS ADDRESS+2 OF LOCATION
690								:TO BE REFERENCED THRU KERNEL PAGE 1
691	003744	052777	000001	176044		BIS	#1,SR0	:TURN ON KT11-D
692	003752	005243			AD25:	INC	-(R3)	:DATIP, DATO TO NR PAGE-SHOULD ABORT
693	003754	042777	000001	176034		BIC	#1,SR0	:TURN OFF KT11-D
694	003762	104006				HLT		:DATIP, DATO TO NR PAGE FAILED TO
695	003764	000432				BR	DONE25	:ABORT
696	003766	017701	176024		RET25:	MOV	SR0,R1	:SAVE CONTENTS OF SRO
697	003772	042777	000001	176016		BIC	#1,SR0	:TURN OFF KT11-D
698	004000	022701	100003			CMP	#100003,R1	:CHECK SAVED CONTENTS OF SRO

699	004004	001401			BEQ	.+4		
700	004006	104006			HLT			:SRO INCORRECT-SHOULD HAVE LOCKED
701								:ON DATO TO KERNEL PAGE 1(NR)
702								:NR FAULT SHOULD BE SET
703	004010	022777	003752	176004	CMP	#AD25, @SR2		:CHECK SR2
704	004016	001401			BEQ	.+4		
705	004020	104006			HLT			:SR2 INCORRECT-SHOULD HAVE LOCKED
706								:ON THE ABORTED REFERENCE, CONTAINING THE
707								:VIRTUAL ADDRESS OF THE INSTRUCTION
708	004022	017704	176044		MOV	@KPDRI, R4		:MOVE CONTENTS OF PDR TO R4
709	004026	042704	000007		BIC	#7, R4		:TO MASK OFF THE ACCESS KEY
710	004032	022704	077400		CMP	#77400, R4		:CHECK PDR
711	004036	001401			BEQ	.+4		:WITH BITS 0-2 MASKED OFF
712	004040	104006			HLT			:KPDRI INCORRECT-SHOULD NOT HAVE
713								:BEEN CHANGED
714	004042	005767	176102		TST	DESTAD		:MAKE CERTAIN THAT DESTINATION
715	004046	001401			BEQ	.+4		:LOCATION WAS NOT WRITTEN
716	004050	104006			HLT			:DATO TO NR PAGE WROTE INTO
717								:THE DESTINATION LOCATION
718	004052	016777	175750	175744	DONE25: MOV	KTSTA, @KTVEC		:CHANGE PAGE FAULT RETURN
719	004060	005077	175742		CLR	@KTSTA		:TO CAUSE A HALT ON AN UNEXPECTED
720	004064	005077	175726		CLR	@SRO		:TRAP
721	004070	005037	177776		CLR	@#PS		
722								:SHOW THAT A DATIP, DATOB SEQUENCE TO A NR PAGE WORD ABORTS
723								:SHOW THAT THE KT11-D STATUS REGISTERS LOCK UP, AND THAT THE PDR
724								:CORRESPONDING TO THE REFERENCE IS CORRECT
725	004074	104400			TEST10: SCOPE			
726	004076	012706	001000		MOV	#KSTACK, SP		:INITIALIZE KERNEL STACK POINTER
727	004102	005077	175710		CLR	@SRO		:INITIALIZE SRO
728	004106	004767	002524		JSR	PC, ORDER		:CHECK TEST SEQUENCE
729	004112	000010			IO			:TEST NUMBER
730	004114	104006			HLT			:TEST EXECUTED OUT OF SEQUENCE
731	004116	004767	001420		JSR	%7, SETUP		:MAKE KERNEL PAGE 1 NR, BANK 0
732								:MAKE KERNEL PAGE 7 RW, EXTERNAL
733								:MAKE ALL OTHER PAGES RW, BANK 0
734	004122	012777	004164	175674	MOV	#RET27, @KTVEC		:SETUP ABORT RETURN
735	004130	005077	175672		CLR	@KTSTA		
736	004134	005067	176010		CLR	DESTAD		:INITIALIZE LOCATION TO BE ADDRESSED
737								:BY DATIP, DATOB TO NR PAGE
738	004140	012704	022150		MOV	#DESTAD+20000, R4		:R4 CONTAINS ADDRESS OF LOCATION
739								:TO BE REFERENCED THRU KERNEL PAGE 1
740	004144	052777	000001	175644	BIS	#1, @SRO		:TURN ON KT11-D
741	004152	105224			AD27: INCB	(R4)+		:DATIP, DATOB TO NR PAGE-SHOULD ABORT
742	004154	005377	175636		DEC	@SRO		:TURN OFF KT11-D
743	004160	104006			HLT			:DATIP, DATO TO NR PAGE FAILED
744	004162	000431			BR	DONE27		:TO ABORT
745	004164	017701	175626		RET27: MOV	@SRO, R1		:SAVE CONTENTS OF SRO
746	004170	005377	175622		DEC	@SRO		:TURN OFF KT11-D
747	004174	022701	100003		CMP	#100003, R1		:CHECK SAVED CONTENTS OF SRO
748	004200	001401			BEQ	.+4		
749	004202	104006			HLT			:SRO INCORRECT-SHOULD HAVE LOCKED ON
750								:DATIP, DATOB TO KERNEL DATA PAGE 1 (NR)
751								:NR FAULT SHOULD BE SET
752	004204	022777	004152	175610	CMP	#AD27, @SR2		:CHECK SR2
753	004212	001401			BEQ	.+4		
754	004214	104006			HLT			:SR2 INCORRECT SHOULD HAVE LOCKED





811	004460	000427				BR	DONE31		
812	004462	017702	175330		OK31:	MOV	SR0,R2		;SAVE CONTENTS OF SRO
813	004466	105377	175324			DECB	SR0		;TURN OFF KT11-D
814	004472	022702	000017			CMP	#17,R2		;CHECK SAVED CONTENTS OF SRO
815	004476	001401				BEQ	.+4		
816	004500	104006				HLT			;SRO INCORRECT-SHOULD HAVE
817									;TRACKED REFERENCE TO
818									;PAGE 0, WHICH GOT THE ADDRESS
819									;OF SRO TO TURN OFF KT11-D
820	004502	022777	004502	175312		CMP	#,SR2		;CHECK SR2
821	004510	001401				BEQ	.+4		
822	004512	104006				HLT			;SR2 INCORRECT-SHOULD TRACK EVEN
823									;WHEN KT11-D IS OFF
824	004514	022777	077406	175350		CMP	#77406,PKPDR1		;CHECK PDR FOR
825	004522	001401				BEQ	.+4		;THE RW PAGE REFERENCED
826	004524	104006				HLT			;KPDR1 INCORRECT-SHOULD NOT
827									;HAVE BEEN CHANGED
828	004526	000404				BR	DONE31		
829	004530	042777	000001	175260	RET31:	BIC	#1,SR0		;TURN OFF KT11-D
830	004536	104006				HLT			;DATI TO RW PAGE CAUSED
831									;A TRAP OR ABORT
832	004540	016777	175262	175256	DONE31:	MOV	KTSTA,KTVEC		;RESTORE TRAP RETURN TO CAUSE HALT
833	004546	005077	175254			CLR	KTSTA		;ON AN UNEXPECTED TRAP
834	004552	005077	175240			CLR	SR0		;INITIALIZE SRO
835	004556	005037	177776			CLR	PS		;INITIALIZE PROCESSOR STATUS
836									
837									
838									;SHOW THAT A DATO (NO DATIP) TO A RW PAGE (ACF=6)
839									;NEITHER TRAPS NOR ABORTS
840									;SHOW THAT THE KT11-D STATUS REGISTERS CONTINUE TO TRACK, AND THAT
841									;THE PDR CORRESPONDING TO THE REFERENCE IS CORRECT
842	004562	104400			TEST12:	SCOPE			
843	004564	012706	001000			MOV	#KSTACK,SP		;INITIALIZE KERNEL STACK POINTER
844	004570	005077	175222			CLR	SR0		;INITIALIZE SRO
845	004574	004767	002036			JSR	PC,ORDER		;CHECK TEST SEQUENCE
846	004600	000012				12			;TEST NUMBER
847	004602	104006				HLT			;TEST EXECUTED OUT OF SEQUENCE
848	004604	012746	000006			MOV	#6,-(SP)		;PUSH RW KEY ON THE STACK
849	004610	004767	000726			JSR	%7,SETUP		;MAKE KERNEL PAGE 1 RW, BANK 0
850									;MAKE KERNEL PAGE 7 RW, EXTERNAL
851	004614	005726				TST	(SP)+		;MAKE ALL OTHER PAGES RW, BANK 0
852	004616	012777	004730	175200		MOV	#RET33,KTVEC		;RESTORE STACK POINTER
853	004624	005077	175176			CLR	KTSTA		;SETUP ABORT RETURN IN CASE
854	004630	005067	175314			CLR	DESTAD		
855	004634	012701	022150			MOV	#DESTAD+20000,R1		;INITIALIZE LOCATION TO BE REFERENCED
856									;R1 CONTAINS VIRTUAL ADDRESS OF
857	004640	005277	175152			INC	SR0		;LOCATION TO BE REFERENCED THRU KERNEL PAGE 1
858	004644	012721	125252			MOV	#125252,(R1)+		;TURN ON KT11-D
859	004650	017702	175142			MOV	SR0,R2		;DATO TO RW PAGE-SHOULDN'T TRAP OR ABORT
860	004654	105377	175136			DECB	SR0		;SAVE CONTENTS OF SRO
861	004660	022702	000017			CMP	#17,R2		;TURN OFF KT11-D
862	004664	001401				BEQ	.+4		;CHECK SAVED CONTENTS OF SRO
863	004666	104006				HLT			;SRO INCORRECT-SHOULD HAVE
864									;TRACKED REFERENCE TO DATA SPACE,
865									;PAGE 0, WHICH GOT THE ADDRESS
866									;OF SRO TO TURN OFF KT11-D



923	005110	104006			HLT				; KPDR1 INCORRECT - "W" BIT SHOULD BE SET
924	005112	022767	000001	175030	CMP	#1, DESTAD			; MAKE CERTAIN THAT THE INSTRUCTION WAS EXECUTED
925	005120	001401			BEQ	.+4			
926	005122	104006			HLT				; DATIP, DATO TO RW PAGE DIDN'T EXECUTE CORRECTLY
927	005124	000404			BR	DONE35			
928	005126	042777	000001	174662	RET35: BIC	#1, @SR0			; TURN OFF KT11-D
929	005134	104006			HLT				; DATIP, DATO TO RW PAGE CAUSED
930									; A TRAP OR ABORT
931	005136	016777	174664	174660	DONE35: MOV	KTSTA, @KTVEC			; RESTORE TRAP RETURN TO CAUSE HALT
932	005144	005077	174656		CLR	@KTSTA			; ON AN UNEXPECTED TRAP
933	005150	005077	174642		CLR	@SR0			; INITIALIZE SR0
934	005154	005037	177776		CLR	@PS			; INITIALIZE PROCESSOR STATUS
935									
936									
937									
938									
939									
940									
941	005160	104400							
942	005162	012706	001000		MOV	#KSTACK, SP			; INITIALIZE KERNEL STACK POINTER
943	005166	005077	174624		CLR	@SR0			; INITIALIZE SR0
944	005172	004767	001440		JSR	PC, ORDER			; CHECK TEST SEQUENCE
945	005176	000014			14				; TEST NUMBER
946	005200	104006			HLT				; TEST EXECUTED OUT OF SEQUENCE
947	005202	012746	000006		MOV	#6, -(SP)			; PUSH RW KEY ON THE STACK
948	005206	004767	000330		JSR	%7, SETUP			; MAKE KERNEL PAGE 1 RW, BANK 0
949									; MAKE KERNEL PAGE 7 RW, EXTERNAL
950									; MAKE ALL OTHER PAGES RW, BANK 0
951	005212	005726			TST	(SP)+			; RESTORE STACK POINTER
952	005214	012777	005324	174602	MOV	@RET37, @KTVEC			; SETUP ABORT RETURN IN CASE
953	005222	005077	174600		CLR	@KTSTA			
954	005226	005067	174716		CLR	DESTAD			; INITIALIZE LOCATION TO BE REFERENCED
955	005232	012703	022151		MOV	#DESTAD+20001, R3			; R3 CONTAINS VIRTUAL ADDRESS+1 OF
956									; LOCATION TO BE REFERENCED THRU KERNEL PAGE 1
957	005236	005277	174554		INC	@SR0			; TURN ON KT11-D
958	005242	105343			DECB	-(R3)			; DATIP, DATOB TO RW PAGE-SHOULDN'T TRAP OR ABORT
959	005244	017702	174546		MOV	@SR0, R2			; SAVE CONTENTS OF SR0
960	005250	105377	174542		DECB	@SR0			; TURN OFF KT11-D
961	005254	022702	000017		CMP	#17, R2			; CHECK SAVED CONTENTS OF SR0
962	005260	001401			BEQ	.+4			
963	005262	104006			HLT				; SR0 INCORRECT-SHOULD HAVE
964									; TRACKED REFERENCE TO DATA SPACE,
965									; PAGE 0, WHICH GOT THE ADDRESS
966									; OF SR0 TO TURN OFF KT11-D
967	005264	022777	005264	174530	CMP	#, @SR2			; CHECK SR2
968	005272	001401			BEQ	.+4			
969	005274	104006			HLT				; SR2 INCORRECT-SHOULD TRACK EVEN
970									; WHEN KT11-D IS OFF
971	005276	022777	077506	174566	CMP	#77506, @KPDR1			; CHECK PDR CORRESPONDING
972	005304	001401			BEQ	.+4			; TO THE RW REFERENCE
973	005306	104006			HLT				; KPDR1 INCORRECT - "W" BIT SHOULD BE SET
974	005310	022767	000377	174632	CMP	#377, DESTAD			; MAKE CERTAIN THAT THE INSTRUCTION WAS EXECUTED
975	005316	001401			BEQ	.+4			
976	005320	104006			HLT				; DATIP, DATOB TO RW PAGE DIDN'T EXECUTE CORRECTLY
977	005322	000404			BR	DONE37			
978	005324	042777	000001	174464	RET37: BIC	#1, @SR0			; TURN OFF KT11-D

; SHOW THAT A DATIP, DATOB SEQUENCE TO A RW PAGE (ACF=6)  
; NEITHER TRAPS NOR ABORTS  
; SHOW THAT THE KT11-D STATUS REGISTERS CONTINUE TO TRACK, AND THAT  
; THE PDR CORRESPONDING TO THE REFERENCE IS CORRECT  
TEST14: SCOPE





```

1000
1001
1002 005414 052113 030461 042055
1003 005422 040440 041503 051505
1004 005430 020123 042513 051531
1005 005436 052040 051505 026124
1006 005444 046440 026504 030461
1007 005452 042055 045502 041124
1008 005460 041055 100
1009 005463 120 036503 040040
1010 005470 020040 051520 020075
1011 005476 100
1012 005500
1013
1014 005500 005077 174312
1015 005504 012701 002030
1016 005510 012700 000010
1017 005514 005071 000020
1018 005520 012731 077406
1019 005524 077005
1020 005526 062701 000020
1021 005532 020127 002126
1022 005536 003764
1023 005540 000207
1024
1025
1026
1027
1028
1029
1030 005542 004767 177732
1031 005546 012777 077400 174316
1032 005554 056677 000002 174310
1033 005562 012777 007600 174336
1034 005570 000207
1035
1036
1037
1038
1039
1040
1041 005572 005037 177776
1042 005576 012706 001000
1043 005602 012737 140000 177776
1044 005610 012706 002000
1045 005614 005037 177776
1046 005620 000000
1047 005622 016767 171742 000036
1048 005630 062767 000002 000030
1049 005636 000000
1050 005640 005067 000126
1051 005644 012767 005656 000122
1052 005652 000177 000010
1053 005656 005067 000110
1054 005662 000177 000000
1055 005666 000000

```

```

;MESSAGE AREA
MTIT: .ASCII 'KT11-D ACCESS KEYS TEST, MD-11-DBKTB-B'

```

```

MPC: .ASCII 'PC= 2'
MPS: .ASCII 'PS= 2'

```

```

;SUBROUTINE TO MAKE ALL PAGES RW, BANK 0, 4K, UP
EVEN
RWALL: CLR 2SR0
MOV #ADRTAB,R1
RWL1: MOV #10,R0
RWL2: CLR 220(R1)
MOV #77406,2(R1)+
SOB R0,RWL2
ADD #20,R1
CMP R1,#ADREND
BLE RWL1
RTS %7

```

```

;SUBROUTINE TO SET ALL PAGES RW EXCEPT KERNEL PAGE 1
;KERNEL PAGE 1 IS SET TO DESIRED KEY
;KEY IS PASSED VIA THE STACK
;ALL PAGES ARE MAPPED TO BANK 0 EXCEPT KERNEL PAGE 7, WHICH IS MAPPED TO
;THE EXTERNAL BANK

```

```

SETUP: JSR %7,RWALL ;INITIALLY MAP ALL PAGES RW, BANK 0
MOV #77400,2KPDRI ;MAKE KERNEL PAGE ONE 4K, UP
BIS 2(SP),2KPDRI ;SET TO DESIRED KEY
MOV #7600,2KPAR7 ;MAP KERNEL PAGE 7 EXTERNAL
RTS %7

```

```

;ROUTINE TO LOOP THRU A SINGLE INSTRUCTION TEST
;LOAD THE STARTING ADDRESS OF THE TEST
;YOU WISH TO RUN (THE ADDRESS OF THE TESTX
;TAG) AT THE 1ST HALT, SET SWITCH REGISTER
;OPTIONS AT THE 2ND HALT.
;NOTE THAT SW11 MUST BE DOWN AFTER THE 2ND HALT

```

```

TESTX: CLR 2#PS
MOV #KSTACK,SP
MOV #140000,2#PS ;SETUP USER STACK POINTER
MOV #USTACK,SP
CLR 2#PS
HALT ;WAIT FOR STARTING ADDRESS
MOV SR,RETRNX ;LOAD STARTING ADDRESS IN RETRNX
ADD #2,RETRNX ;ADD 2 TO POINT TO INSTRUCTION AFTER
HALT ;SET SR OPTIONS
CLR SCOPEF ;KEEP COUNT AT ZERO
MOV #XLOOP,RETURN ;LOAD SCOPE LOOP RETURN POINTER
JMP 2RETRNX ;JUMP TO TEST
XLOOP: CLR SCOPEF ;KEEP COUNT AT ZERO
JMP 2RETRNX ;JUMP TO TEST
RETRNX: 0

```

```

1056
1057
1058 005670 032737 040000 177570 ;SCOPE AND/OR ITERATION LOOP FOR EACH TEST 4000 TIMES
SCOPEC: BIT #40000,2#SR ;TEST SR FOR SCOPE
1059 005676 001020 BNE SCOPEB ;YES SCOPE
1060 005700 005767 174224 TST PASCNT ;FIRST PASS?
1061 005704 001422 BEQ SCOPEG ;YES, SKIP ITERATIONS
1062 005706 032737 004000 177570 BIT #4000,2#SR ;NO-TEST FOR ITERATION
1063 005714 001016 BNE SCOPEG ;INHIBIT ITERATION
1064 005716 026767 000050 000044 CMP SCOPEF, ICOUNT ;COMPARE CURRENT COUNT TO MAX NUMBER
1065 005724 100012 BPL SCOPEG ;EXIT-DONE
1066 005726 005267 000040 INC SCOPEF ;INCREMENT COUNT
1067 005732 012737 000340 177776 MOV #340,2#PS ;PREVENT TRAPPING WHILE MOVING STACK
SCOPEB: CMP (6)+,%6 ;REPOSITION STACK
1068 005740 022606 MOV (6)+,2#PS ;RESTORE PREVIOUS PROCESSOR STATUS
1069 005742 012637 177776 JMP @RETURN ;REPEAT TEST
SCOPEG: CLR SCOPEF ;CLEAR COUNT
1071 005752 005067 000014 INC TESTCT ;STEP TEST COUNTER
1072 005756 005367 000740 MOV @%6,RETURN ;SAVE SCOPE RETURN POINTER
1073 005762 011667 000006 RTI ;RETURN INLINE-NEXT TEST
1074 005766 000002 ICOUNT: 4000 ;ITERATION COUNT
1075 005770 004000 SCOPEF: 0 ;COUNT LOCATION FOR ITERATION LOOP
1076 005772 000000 RETURN: 0 ;ADDRESS OF LAST TEST
1077 005774 000000
1078
1079
1080
1081 ;ENTERED WITH SYSTEM TRAP CALL (HLT)
1082 005776 012767 000340 171772 ;PRINT OUT THE ERROR PC+2 AND STATUS REGISTER
PRINT: MOV #340,PS ;SET PRIORITY TO 7
1083 006004 036727 171560 020000 BIT SR,#20000 ;TEST FOR INHIBIT PRINT OUT
1084 006012 001401 BEQ .+4 ;BRANCH TO PRINT
1085 006014 000432 BR CK ;INHIBIT, CHECK FOR HALT
1086 006016 012667 000102 MOV (6)+,SAVPC ;PC OF FAILING ROUTINE
1087 006022 012667 000100 MOV (6)+,SAVPSR ;PSR OF ERROR CONDITION
1088 006026 024646 CMP -(6),-(6) ;RESTORE STACK
1089 006030 012767 000200 171740 MOV #200,PS
1090 006036 004767 000446 JSR %7,CRLF ;OUTPUT CARRIAGE RETURN AND LINE FEED
1091 006042 016767 000056 000324 MOV SAVPC,PTEMP1 ;LOAD WITH FAILING PC+2
1092 006050 004767 000502 JSR PC,TYPE
1093 006054 005463 MPC
1094 006056 004767 000046 JSR PC,PRSHRT
1095 006062 004767 000470 JSR PC,TYPE
1096 006066 005470 MPS
1097 006070 016767 000032 000276 MOV SAVPSR,PTEMP1 ;LOAD PROCESSOR STATUS
1098 006076 004767 000060 JSR %7,PROCT ;PRINT PROCESSOR STATUS
1099 006102 023737 000042 000046 CK: CMP @#42,@#46 ;ARE WE IN ACT11 AUTOMATIC MODE?
1100 006110 001403 BEQ .+10 ;YES, HALT ON ERROR
1101 006112 005767 171452 TST SR ;CHECK SR FOR HALT SWITCH
1102 006116 100001 BPL .+4 ;BRANCH IF NOT SET
1103 006120 000000 HALT ;HALT ON ERROR UP
1104 006122 000002 RTI ;RETURN TO MAIN LINE
1105 006124 000000 SAVPC: 0
1106 006126 000000 SAVPSR: 0

```

```

1107
1108 ;SUBROUTINE TO PRINT OUT OCTAL NUMBER
1109 ;PRSHRT DELETES LEADING ZEROS
1110 ;PROCT PRINTS OUT 6 OCTAL DIGITS
1111 006130 012767 000001 000232 PRSHRT: MOV #1, PRSFLG ;SET FLAG TO INDICATE SHORT PRINTOUT
1112 006136 005767 000232 TST PTEMP1 ;CHECK FOR ZERO
1113 006142 001011 BNE PROCT+4 ;BRANCH IF NOT ZERO
1114 006144 012777 000260 173642 MOV #260, @TDBR ;OUTPUT A SINGLE ZERO
1115 006152 105777 173634 TSTB @TCSR ;WAIT FOR TTY READY
1116 006156 100375 BPL .-4
1117 006160 000207 RTS ;RETURN
1118 006162 005067 000202 PROCT: CLR PRSFLG ;CLEAR FLAG TO INDICATE FULL PRINTOUT
1119 006166 005067 000206 CLR PTEMP3 ;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
1120 006172 005067 000174 CLR PRFLG ;INITIALIZE CARRY FLAG FOR ROTATES
1121 006176 012767 000260 000172 MOV #260, PTEMP2 ;SETUP R3
1122 006204 005767 000164 TST PTEMP1 ;CHECK BIT 15 OF NUMBER
1123 006210 100002 BPL .+6 ;BRANCH IF ZERO
1124 006212 005267 000160 INC PTEMP2 ;INCREMENT R3 IF ONE
1125 006216 006167 000152 ROL PTEMP1 ;ROTATE LEFT MOST OCTAL TO RIGHT END
1126 006222 006167 000146 ROL PTEMP1
1127 006226 005567 000140 ADC PRFLG
1128 006232 005767 000132 P.CK: TST PRSFLG ;STORE CARRY
1129 006236 001404 BEQ P.WAIT ;CHECK FOR SHORT PRINTOUT
1130 006240 026727 000132 000260 CMP PTEMP2, #260 ;BRANCH IF NOT SET
1131 006246 001410 BEQ P.CONT ;CHECK FOR ZERO IF SET
1132 006250 016777 000122 173536 P.WAIT: MOV PTEMP2, @TDBR ;IF SET, GO TO NEXT CHARACTER
1133 006256 105777 173530 TSTB @TCSR ;OUTPUT NEXT CHARACTER
1134 006262 100375 BPL .-4 ;WAIT FOR TTY READY
1135 006264 005067 000100 CLR PRSFLG ;PRINT REST OF NUMBER AFTER A NON-ZERO DIGIT
1136 006270 005267 000104 P.CONT: INC PTEMP3 ;COUNT
1137 006274 026727 000100 000006 CMP PTEMP3, #6 ;CHECK FOR DONE
1138 006302 001001 BNE P.CNT1 ;BRANCH IF NOT DONE
1139 006304 000207 RTS
1140 006306 000241 P.CNT1: CLC ;CLEAR CARRY
1141 006310 005767 000056 TST PRFLG ;CHECK FOR PREVIOUS CARRY
1142 006314 001403 BEQ .+10 ;BRANCH IF PREVIOUSLY ZERO
1143 006316 005067 000050 CLR PRFLG ;INITIALIZE FLAG
1144 006322 000261 SEC ;SET CARRY
1145 006324 006167 000044 ROL PTEMP1 ;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTER
1146 006330 006167 000040 ROL PTEMP1
1147 006334 006167 000034 ROL PTEMP1
1148 006340 005567 000026 ADC PRFLG
1149 006344 016767 000024 000024 MOV PTEMP1, PTEMP2 ;STORE CARRY
1150 006352 042767 177770 000016 BIC #177770, PTEMP2 ;LOAD DATA INTO R3
1151 006360 052767 000260 000010 BIS #260, PTEMP2 ;CLEAR ALL BUT LOWEST OCTAL DIGIT
1152 006366 000721 BR P.CK ;SET TO ASCII EQUIVALENT
1153 006370 000000 PRSFLG: 0 ;LOOP
1154 006372 000000 PRFLG: 0
1155 006374 000000 PTEMP1: 0 ;CONTAINS VALUE TO BE OUTPUT
1156 006376 000000 PTEMP2: 0 ;SCRATCH
1157 006400 000000 PTEMP3: 0 ;USED TO COUNT CHARACTERS OUTPUT

```

```

1158
1159
1160
1161 006402 011667 000032
1162 006406 162767 000002 000024
1163 006414 017767 000020 000016
1164 006422 105067 000013
1165 006426 062767 006442 000004
1166 006434 017707 000000
1167 006440 000000
1168 000000
1169 000000
1170 000000
1171 006442 000000
1172 006444 000000
1173 006446 000000
1174 006450 005776
1175
1176
1177
1178 006452 012777 000207 173334
1179 006460 105777 173326
1180 006464 100375
1181 006466 012777 000052 173320
1182 006474 105777 173312
1183 006500 100375
1184 006502 004767 000002
1185 006506 000207
1186
1187
1188 006510 012777 000215 173276
1189 006516 105777 173270
1190 006522 100375
1191 006524 012777 000212 173262
1192 006532 105777 173254
1193 006536 100375
1194 006540 012777 000177 173246
1195 006546 105777 173240
1196 006552 100375
1197 006554 000207

; EMT HANDLER
; FIRST 3 CALLS LEFT OPEN IN TABLE FOR EASY PATCHES
EMTSRV: MOV @SP, EPC ; GET CALL
SUB #2, EPC
MOV @EPC, EPC
CLRB EPC+1 ; SAVE OFFSET ONLY
ADD @EMTAB, EPC ; POINT TO TABLE OF ADDRESSES
MOV @EPC, PC ; JUMP TO DESIRED ROUTINE

EPC: 0
PATCH1=0 ; SUBSTITUTE 104000 WHERE 1ST PATCH IS NEEDED
PATCH2=0 ; 104002 FOR 2ND PATCH
PATCH3=0 ; 104004 FOR 3RD PATCH
EMTAB: PATCH1 ; LOAD ADDRESS OF 1ST PATCH HERE
PATCH2 ; LOAD ADDRESS OF 2ND PATCH HERE
PATCH3 ; LOAD ADDRESS OF 3RD PATCH HERE
PRINT

; BELL AND '*' ON PASS COMPLETE
BELL: MOV #207, @TDBR
TSTB @TCSR
BPL .-4
MOV #52, @TDBR ; OUTPUT '*'
TSTB @TCSR
BPL .-4
JSR %7, CRLF ; DO A CR AND LF
RTS %7

; SUBROUTINE TO OUTPUT CARRIAGE RETURN AND LINEFEED
CRLF: MOV #215, @TDBR ; OUTPUT CARRIAGE RETURN
TSTB @TCSR ; WAIT FOR TTY READY
BPL .-4
MOV #212, @TDBR ; OUTPUT LINEFEED
TSTB @TCSR ; WAIT FOR TTY READY
BPL .-4
MOV #177, @TDBR ; OUTPUT A NULL CHARACTER
TSTB @TCSR
BPL .-4
RTS %7 ; RETURN

```

```

1198
1199
1200 006556 010067 000052
1201 006562 011600
1202 006564 062716 000002
1203 006570 011000
1204 006572 112067 000034
1205 006576 122767 000100 000026
1206 006604 001003
1207 006606 016700 000022
1208 006612 000207
1209 006614 116777 000012 173172
1210 006622 105777 173164
1211 006626 100375
1212 006630 000760
1213 006632 000000
1214 006634 000000
1215
1216
1217 006636 005037 177776
1218 006642 011667 000052
1219 006646 017767 000046 000044
1220 006654 032737 002000 177570
1221 006662 001404
1222 006664 016700 000030
1223 006670 000005
1224 006672 000000
1225 006674 026767 000022 000016
1226 006702 001403
1227 006704 062716 000002
1228 006710 000207
1229 006712 062716 000004
1230 006716 000207
1231 006720 000000
1232 006722 000000
1233 000001

;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE
↑TYPE: MOV %0, SAVRO
MOV (6), %0 ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
ADD #2, %6 ;SET UP EXIT
MOV @%0, %0
TYP A: MOV B (0), TYPDAT ;GET CHARACTER
CMP B #100, TYPDAT ;CHECK FOR "0" CHARACTER
BNE TYPB ;BRANCH IF NOT "0"
MOV SAVRO, %0 ;RESTORE RO
RTS PC ;TERMINATOR CHAR. EXIT
TYP B: MOV B TYPDAT, @TDBR ;OUTPUT CHAR TO PRINTER
TST B @TCSR ;WAIT FOR TTY READY
BPL -4
BR TYP A
TYPDAT: 0
SAVRO: 0

;SUBROUTINE TO CHECK TEST SEQUENCE
ORDER: CLR @#PS ;CLEAR PROCESSOR STATUS
MOV (SP), TEMPN ;GET TEST NUMBER ADDRESS
MOV @TEMPN, TEMPN ;GET TEST NUMBER
BIT #2000, @#SR
BEQ ORDERB
MOV TEMPN, RO
RESET
HALT
ORDERB: CMP TESTCT, TEMPN ;IS TEST SEQUENCE CORRECT
BEQ ORDERA ;YES, CONTINUE
ADD #2, (SP) ;UPDATE FOR ERROR RETURN
RTS PC
ORDERA: ADD #4, (SP) ;UPDATE FOR GOOD RETURN
RTS PC
TEMPN: 0
TESTCT: 0
.END

```



ADREND=	002126	KPAR0	002110	PATCH2=	000000	R6	=%000006	TEST3	002670
AORTAB	002030	KPAR1	002112	PATCH3=	000000	R7	=%000007	TEST4	003070
AD21	003360	KPAR2	002114	PRFLG	006372	SAVPC	006124	TEST5	003264
AD23	003552	KPAR3	002116	PRINT	005776	SAVPSR	006126	TEST6	003474
AD25	003752	KPAR4	002120	PROCT	006162	SAVRO	006634	TEST7	003674
AD27	004152	KPAR5	002122	PRSFLG	006370	SCOPE =	104400	TYP A	006572
AD4	002556	KPAR6	002124	PRSHRT	006130	SCOPEB	005740	TYPB	006614
AD5	002754	KPAR7	002126	PS =	177776	SCOPEC	005670	TYPDAT	006632
AD6	003154	KPDR0	002070	PTEMP1	006374	SCOPEF	005772	TYPE	006556
BELL	006452	KPDR1	002072	PTEMP2	006376	SCOPEG	005752	UPAR0	002050
CK	006102	KPDR2	002074	PTEMP3	006400	SETUP	005542	UPAR1	002052
CMPOK1	002372	KPDR3	002076	P.CK	006232	SR =	177570	UPAR2	002054
CRLF	006510	KPDR4	002100	P.CNT1	006306	SRO	002016	UPAR3	002056
DESTAD	002150	KPDR5	002102	P.CONT	006270	SROH	002134	UPAR4	002060
DONE1	002450	KPDR6	002104	P.WAIT	006250	SR1	002020	UPAR5	002062
DONE21	003452	KPDR7	002106	RERUNA	003312	SR1H	002136	UPAR6	002064
DONE23	003652	KSTACK	001000	RETRN	005666	SR2	002022	UPAR7	002066
DONE25	004052	KTSTA	002026	RETURN	005774	SR2H	002140	UPDR0	002030
DONE27	004246	KTVEC	002024	RET1	002440	START	002152	UPDR1	002032
DONE31	004540	MPC	005463	RET21	003372	STATUS=	177776	UPDR2	002034
DONE33	004740	MPS	005470	RET23	003570	TCSR	002012	UPDR3	002036
DONE35	005136	MTIT	005414	RET25	003766	TDBR	002014	UPDR4	002040
DONE37	005334	NOP =	000240	RET27	004164	TEMPN	006720	UPDR5	002042
DONE4	002646	NRCNT	002142	RET31	004530	TESTCT	006722	UPDR6	002044
DONE5	003046	NRKEYS	002144	RET33	004730	TESTN =	000015	UPDR7	002046
DONE6	003242	NXTST	004342	RET35	005126	TESTX	005572	USTACK	002000
EMTAB	006442	NXTST1	004360	RET37	005324	TEST1	002270	XLOOP	005656
EMTSRV	006402	OK31	004462	RET4	002572	TEST10	004074	SENDAD	005400
END	005410	ORDER	006636	RET5	002770	TEST11	004360	.	= 006724
EPC	006440	ORDERA	006712	RET6	003166	TEST12	004562		
FTITLE	002132	ORDERB	006674	RWALL	005500	TEST13	004762		
HLT =	104006	PASCNT	002130	RWL1	005510	TEST14	005160		
ICOUNT	005770	PATCH1=	000000	RWL2	005514	TEST2	002472		

. ABS. 006724 000

ERRORS DETECTED: 0  
DEFAULT GLOBALS GENERATED: 0

MULE:DBKTB8,MULE:DBKTB8/SOL=DSKZ:SYSMAC.SML,MULE:DBKTB8.P11  
RUN-TIME: 7 8 .1 SECONDS  
RUN-TIME RATIO: 122/16=7.6  
CORE USED: 31K (61 PAGES)