

B1

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

.REM &

IDENTIFICATION

PRODUCT CODE: AC-T706B-MC
PRODUCT NAME: CZKDKB0 KDJ11 MEMORY MANAGEMENT DIAGNOSTIC
PRODUCT DATE: 15-MAR-84
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHORS: HENRY ENMAN, JIM PITTMAN, BARRY IRRGANG

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.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1984 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL PDP VIBUS MASSBUS
DEC DECUS OF TAPE

&

39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55

HISTORY

.REM 8

OCT 83 REV. A
FEB-84 REV. B

- FIRST RELEASE
- CORRECTIONS MADE TO:
1. CORRECT VECTOR AREA MAINTENANCE PROBLEM
 2. PREVENT \$TESTN FROM GETTING OUT OF SYNC WHEN SKIPPING Deselected TESTS.
 3. TURN CACHE MEMORY SYSTEM OFF DURING NON-CACHE TESTS.
 4. ENSURE THAT CPU ERROR REGISTER IS CLEARED AFTER COMPLETION OF TEST THAT MIGHT CAUSE IT TO BE SET.
 5. SAVE PC AND CONTENTS OF R6 ON UNEXPECTED INTERRUPTS
- ADDITIONAL TESTS TO IMPROVE TEST COVERAGE INCLUDE:
1. NON-EXISTANT MEMORY TRAP TEST

&

D1

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

.REM &

TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
2.1	LOADING AND STARTING PROCEDURE
2.2	PROGRAM OPTIONS
2.3	OPERATION UNDER APT
3.0	ERROR INFORMATION
4.0	PROGRESS REPORT

&

.REM 6

79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS IS AN APT COMPATIBLE VERSION OF THE KDJ11 MEMORY MANAGEMENT DIAGNOSTIC. IT FOCUSES ON TESTING THE FUNCTIONALITY OF THE MEMORY MANAGEMENT FEATURES. THE TEST REQUIRES 4 MEGABYTES OF QBUS MEMORY TO FULLY TEST THE MMU ADDER. A SUBSET OF THE ADDER IS TESTED IF LESS THAN 4 MEGABYTES OF MEMORY IS AVAILABLE (MINIMUM OF 28 KBYTES). IN ADDITION, FOR TESTING IN QBUS SYSTEMS WITH ONLY 18 ADDRESS BITS, A MEANS IS PROVIDED TO SKIP TESTS WHICH REQUIRE 22 BIT ADDRESSES. THIS FEATURE IS IMPLEMENTED BY SETTING BIT 08 IN THE SOFTWARE SWITCH REGISTER (LOCATION 176) TO A ONE. DEFAULT IS TO TEST 22 BIT ADDRESSES.

1.2 SYSTEM REQUIREMENTS

KDJ11-A PROCESSOR MODULE
ENSURE THAT HALT TRAP OPTION IS DISABLED (JUMPER W9 INSTALLED)
32KW MEMORY
Q-22 BACKPLANE (18 BIT QBUS MAY BE USED WITH REDUCED TEST COVERAGE)
SERIAL LINE UNIT AND CONSOLE TERMINAL (CONSOLE TERMINAL NOT REQUIRED FOR APT)

1.3 RELATED DOCUMENTS AND STANDARDS

KDJ11-A MODULE SPECIFICATION REV 2.2
PDP11 MAINDEC SYSMAC PACKAGE
J11 CONTROL CHIP SPECIFICATION 21-17679-00
J11 DATA CHIP SPECIFICATION 21-17677-00

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE KDJ11 CPU DIAGNOSTIC MUST RUN SUCCESSFULLY PRIOR TO RUNNING THE MEMORY MANAGEMENT TEST.

1.5 ASSUMPTIONS

IT IS ASSUMED THAT THE DIAGNOSTIC OPERATOR IS FAMILIAR WITH THE XXDP, OPERATING SYSTEM AND THE J11 MICRO-ODT.

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURE

LOAD PROGRAM INTO MEMORY USING STANDARD XXDP, PROCEDURES. THE PROGRAM IS STARTED BY LOADING ADDRESS 200 AND USING THE J11 MICRO-ODT G COMMAND TO START. THE PROGRAM IDENTIFICATION MESSAGE WILL BE TYPED AFTER THE FIRST PASS OF THE COMPLETE PROGRAM.

2.2 PROGRAM OPTIONS

135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184

THE FOLLOWING ASSIGNMENTS HAVE BEEN MADE FOR THE KDJ11-A
DIAGNOSTIC SWITCH REGISTER BITS:

BIT#15	14	13	12	11	10	9	8	
						DON'T	18 BIT	EXTENDED
						TEST	ADDRESS	CACHE
						BEVENT	ONLY	TESTS

DEFAULT SETTINGS ARE TO TEST 22 BIT ADDRESSES. THE OTHER BITS HAVE
NO EFFECT ON THE OPERATION OF THE PROGRAM.

PRIOR TO EXECUTING THE FIRST PASS OF THE DIAGNOSTIC THE OPERATOR
WILL BE DIRECTED TO SET THE SWITCH REGISTER TO INDICATE WHETHER
THE KDJ11-A UNDER TEST IS IN A SYSTEM CONFIGURED FOR 18 OR 22 BIT
ADDRESSING. AN 18 BIT ADDRESS CONFIGURATION SHOULD BE INDICATED IF
ANY 18 ADDRESS BIT ONLY MEMORY BOARDS RESIDE IN THE SYSTEM OR IF
THE SYSTEM BACKPLANE DOES NOT SUPPORT 22 ADDRESS BITS.

TO CHANGE THE SWITCH REGISTER, HALT THE PROGRAM, AND EITHER RESTART
THE PROGRAM AT 200 ANSWERING THE INITIAL QUESTIONS, OR LOAD THE
SOFTWARE SWITCH REGISTER (ADDRESS 176) WITH THE DESIRED OPTIONS AND
RESTART THE PROGRAM USING THE J11 MICRO ODT P COMMAND.

2.3 OPERATION UNDER APT

THERE ARE NO DIFFERENCES IN THE EXECUTION OF THIS DIAGNOSTIC
WHEN OPERATING IN AN APT ENVIRONMENT. PROBLEMS CAUSED BY THE
ASYNCHRONOUS HALTS OF THE DIAGNOSTIC BY THE APT MONITOR HAVE
NOT BEEN NOTED.

3.0 ERROR INFORMATION

ERRORS WILL CAUSE THE FOLLOWING ERROR MESSAGE TO BE PRINTED:

```
ERROR DURING MMU TESTING  
ERROR # * (UNIQUE ERROR NUMBER)  
ERROR PC * (PC AT TIME OF ERROR)
```

THE ERROR WILL THEN BE REPORTED TO APT AND THE PROGRAM
WILL HALT.

4.0 PROGRESS REPORT

AT THE END OF EACH PASS THE DIAGNOSTIC NAME AND PASS COUNT ARE PRINTED.

ABASE	=	000000	576			
ABORT0		012214	2383	2509		
ABORT1		012402	2394	2571		
ACDW1	=	000000	576			
ACDW2	=	000000	576			
ACPUOP	=	000000	576	591		
ADDTRP		001454	736	840	869	
ADDW0	=	000000	576			
ADDW1	=	000000	576			
ADDW10	=	000000	576			
ADDW11	=	000000	576			
ADDW12	=	000000	576			
ADDW13	=	000000	576			
ADDW14	=	000000	576			
ADDW15	=	000000	576			
ADDW2	=	000000	576			
ADDW3	=	000000	576			
ADDW4	=	000000	576			
ADDW5	=	000000	576			
ADDW6	=	000000	576			
ADDW7	=	000000	576			
ADDW8	=	000000	576			
ADDW9	=	000000	576			
ADEVCT	=	000000	576	582		
ADEVH	=	000000	576			
AENV	=	000000	576	587		
AENVH	=	000000	576	588		
AFATAL	=	000000	576	579		
ALLCTR		001056	618			
AMADR1	=	000000	576			
AMADR2	=	000000	576			
AMADR3	=	000000	576			
AMADR4	=	000000	576			
AMAMS1	=	000000	576			
AMAMS2	=	000000	576			
AMAMS3	=	000000	576			
AMAMS4	=	000000	576			
AMSGAD	=	000000	576	584		
AMSGLG	=	000000	576	585		
AMSGTY	=	000000	576	578		
AMTYP1	=	000000	576			
AMTYP2	=	000000	576			
AMTYP3	=	000000	576			
AMTYP4	=	000000	576			
APASS	=	000000	576	581		
APRIOR	=	000000	576			
APTCSU	=	000040	3749	4234		
APTENV	=	000001	3742	4190	4232	4261
APTSIZ	=	000200	801	4231		
APTSP0	=	000100	3744	4192	4233	
ASWREG	=	002000	495	576	589	
ATESTN	=	000000	576	580		
AUNIT	=	000000	576	583		
AUSWR	=	000000	576	590		
AVECT1	=	000000	576			
AVECT2	=	000000	576			

```

206 .TITLE GLOBAL AREAS
207 .SBTTL GLOBAL EQUATES SECTION
208
209 ;**
210 ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
211 ; ARE USED IN MORE THAN ONE TEST.
212 ;**
213 .SBTTL BASIC DEFINITIONS
214
215 ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
216 001000 STACK= 1000
217 .EQUIV FMT,ERROR ;;BASIC DEFINITION OF ERROR CALL
218 .EQUIV IOT,SCOPE ;;BASIC DEFINITION OF SCOPE CALL
219
220 ;*MISCELLANEOUS DEFINITIONS
221 000011 HI= 11 ;;CODE FOR HORIZONTAL TAB
222 000012 LF= 12 ;;CODE FOR LINE FEED
223 000015 CR= 15 ;;CODE FOR CARRIAGE RETURN
224 000200 CRLF= 200 ;;CODE FOR CARRIAGE RETURN-LINE FEED
225 177776 PS= 177776 ;;PROCESSOR STATUS WORD
226 .EQUIV PS,PSW
227 177774 STKLMT= 177774 ;;STACK LIMIT REGISTER
228 177772 PIRQ= 177772 ;;PROGRAM INTERRUPT REQUEST REGISTER
229 177570 DSWR= 177570 ;;HARDWARE SWITCH REGISTER
230 177570 DDISP= 177570 ;;HARDWARE DISPLAY REGISTER
231
232 ;*GENERAL PURPOSE REGISTER DEFINITIONS
233 000000 R0= #0 ;;GENERAL REGISTER
234 000001 R1= #1 ;;GENERAL REGISTER
235 000002 R2= #2 ;;GENERAL REGISTER
236 000003 R3= #3 ;;GENERAL REGISTER
237 000004 R4= #4 ;;GENERAL REGISTER
238 000005 R5= #5 ;;GENERAL REGISTER
239 000006 R6= #6 ;;GENERAL REGISTER
240 000007 R7= #7 ;;GENERAL REGISTER
241 000006 SP= #6 ;;STACK POINTER
242 000007 PC= #7 ;;PROGRAM COUNTER
243
244 ;*PRIORITY LEVEL DEFINITIONS
245 000000 PR0= 0 ;;PRIORITY LEVEL 0
246 000040 PR1= 40 ;;PRIORITY LEVEL 1
247 000100 PR2= 100 ;;PRIORITY LEVEL 2
248 000140 PR3= 140 ;;PRIORITY LEVEL 3
249 000200 PR4= 200 ;;PRIORITY LEVEL 4
250 000240 PR5= 240 ;;PRIORITY LEVEL 5
251 000300 PR6= 300 ;;PRIORITY LEVEL 6
252 000340 PR7= 340 ;;PRIORITY LEVEL 7
253
254 ;*"SWITCH REGISTER" SWITCH DEFINITIONS
255 100000 SW15= 100000
256 040000 SW14= 40000
257 020000 SW13= 20000
258 010000 SW12= 10000
259 004000 SW11= 4000
260 002000 SW10= 2000
261 001000 SW09= 1000

```

GLOBAL AREAS MACY11 30A(1052) 20 MAR 84 11:51 PAGE 8
 KDJ11A.MAC 20 MAR 84 11:19 BASIC DEFINITIONS

SEQ 0008

```

262      000400      SW08= 400
263      000200      SW07= 200
264      000100      SW06= 100
265      000040      SW05= 40
266      000020      SW04= 20
267      000010      SW03= 10
268      000004      SW02= 4
269      000002      SW01= 2
270      000001      SW00= 1
271      .EQUIV      SW09,SW9
272      .EQUIV      SW08,SW8
273      .EQUIV      SW07,SW7
274      .EQUIV      SW06,SW6
275      .EQUIV      SW05,SW5
276      .EQUIV      SW04,SW4
277      .EQUIV      SW03,SW3
278      .EQUIV      SW02,SW2
279      .EQUIV      SW01,SW1
280      .EQUIV      SW00,SW0
281
282      ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
283      100000      BIT15= 100000
284      040000      BIT14= 40000
285      020000      BIT13= 20000
286      010000      BIT12= 10000
287      004000      BIT11= 4000
288      002000      BIT10= 2000
289      001000      BIT09= 1000
290      000400      BIT08= 400
291      000200      BIT07= 200
292      000100      BIT06= 100
293      000040      BIT05= 40
294      000020      BIT04= 20
295      000010      BIT03= 10
296      000004      BIT02= 4
297      000002      BIT01= 2
298      000001      BIT00= 1
299      .EQUIV      BIT09,BIT9
300      .EQUIV      BIT08,BIT8
301      .EQUIV      BIT07,BIT7
302      .EQUIV      BIT06,BIT6
303      .EQUIV      BIT05,BIT5
304      .EQUIV      BIT04,BIT4
305      .EQUIV      BIT03,BIT3
306      .EQUIV      BIT02,BIT2
307      .EQUIV      BIT01,BIT1
308      .EQUIV      BIT00,BIT0
309
310      ;*BASIC "CPU" TRAP VECTOR ADDRESSES
311      000004      FRRVEC= 4          ;;TIME OUT AND OTHER ERRORS
312      000010      RESVEC= 10       ;;RESERVED AND ILLEGAL INSTRUCTIONS
313      000014      IBIVEC= 14       ;;I/O BIT
314      000014      TRIVEC= 14       ;;TRACE TRAP
315      000014      BPTVEC= 14       ;;BREAKPOINT TRAP (BPT)
316      000020      IOTVEC= 20       ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
317      000024      PWRVEC= 24       ;;POWER FAIL

```


GLOBAL AREAS MACY11 30A(1052) 20-MAR-84 11:31 PAGE 9
 KDJ11A.MAC 20-MAR-84 11:19 BASIC DEFINITIONS

SEQ 0009

```

318      000030      EMTVEC= 30          ;;EMULATOR TRAP (EMT) **ERRCR**
319      000034      TRAPVEC=34        ;; "TRAP" TRAP
320      000060      TKVEC= 60          ;;TTY KEYBOARD VECTOR
321      000064      IPVEC= 64          ;;TTY PRINTER VECTOR
322      000240      PIRQVEC=240       ;;PROGRAM INTERRUPT REQUEST VECTOR
323
324      .SBTTL      MEMORY MANAGEMENT DEFINITIONS
325
326      ;*KT11 VECTOR ADDRESS
327      000250      MMVEC= 250
328
329      ;*KT11 STATUS REGISTER ADDRESSES
330
331      177572      SR0= 177572
332      177574      SR1= 177574
333      177576      SR2= 177576
334      172516      SR3= 172516
335
336      ;*USER "I" PAGE DESCRIPTOR REGISTERS
337
338      177600      UIPDR0= 177600
339      177602      UIPDR1= 177602
340      177604      UIPDR2= 177604
341      177606      UIPDR3= 177606
342      177610      UIPDR4= 177610
343      177612      UIPDR5= 177612
344      177614      UIPDR6= 177614
345      177616      UIPDR7= 177616
346
347      ;*USER "D" PAGE DESCRIPTOR REGISTERS
348
349      177620      UDPDR0= 177620
350      177622      UDPDR1= 177622
351      177624      UDPDR2= 177624
352      177626      UDPDR3= 177626
353      177630      UDPDR4= 177630
354      177632      UDPDR5= 177632
355      177634      UDPDR6= 177634
356      177636      UDPDR7= 177636
357
358      ;*USER "I" PAGE ADDRESS REGISTERS
359
360      177640      UIPAR0= 177640
361      177642      UIPAR1= 177642
362      177644      UIPAR2= 177644
363      177646      UIPAR3= 177646
364      177650      UIPAR4= 177650
365      177652      UIPAR5= 177652
366      177654      UIPAR6= 177654
367      177656      UIPAR7= 177656
368
369      ;*USER "D" PAGE ADDRESS REGISTERS
370
371      177660      UOPAR0= 177660
372      177662      UOPAR1= 177662
373      177664      UOPAR2= 177664

```

374	177666	UDPAR3= 177666
375	177670	UDPAR4= 177670
376	177672	UDPAR5= 177672
377	177674	UDPAR6= 177674
378	177676	UDPAR7= 177676
379		
380		; *SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
381		
382	172200	SIPDR0= 172200
383	172202	SIPDR1= 172202
384	172204	SIPDR2= 172204
385	172206	SIPDR3= 172206
386	172210	SIPDR4= 172210
387	172212	SIPDR5= 172212
388	172214	SIPDR6= 172214
389	172216	SIPDR7= 172216
390		
391		; *SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
392		
393	172220	SDPDR0= 172220
394	172222	SDPDR1= 172222
395	172224	SDPDR2= 172224
396	172226	SDPDR3= 172226
397	172230	SDPDR4= 172230
398	172232	SDPDR5= 172232
399	172234	SDPDR6= 172234
400	172236	SDPDR7= 172236
401		
402		; *SUPERVISOR "I" PAGE ADDRESS REGISTERS
403		
404	172240	SIPAR0= 172240
405	172242	SIPAR1= 172242
406	172244	SIPAR2= 172244
407	172246	SIPAR3= 172246
408	172250	SIPAR4= 172250
409	172252	SIPAR5= 172252
410	172254	SIPAR6= 172254
411	172256	SIPAR7= 172256
412		
413		; *SUPERVISOR "D" PAGE ADDRESS REGISTERS
414		
415	172260	SDPAR0= 172260
416	172262	SDPAR1= 172262
417	172264	SDPAR2= 172264
418	172266	SDPAR3= 172266
419	172270	SDPAR4= 172270
420	172272	SDPAR5= 172272
421	172274	SDPAR6= 172274
422	172276	SDPAR7= 172276
423		
424		; *KERNEL "I" PAGE DESCRIPTOR REGISTERS
425		
426	172300	KIPDR0= 172300
427	172302	KIPDR1= 172302
428	172304	KIPDR2= 172304
429	172306	KIPDR3= 172306

```

430      172310      KIPDR4= 172310
431      172312      KIPDR5= 172312
432      172314      KIPDR6= 172314
433      172316      KIPDR7= 172316
434
435      ;*KERNEL "D" PAGE DESCRIPTOR REGISTERS
436
437      172320      KDPDR0= 172320
438      172322      KDPDR1= 172322
439      172324      KDPDR2= 172324
440      172326      KDPDR3= 172326
441      172330      KDPDR4= 172330
442      172332      KDPDR5= 172332
443      172334      KDPDR6= 172334
444      172336      KDPDR7= 172336
445
446      ;*KERNEL "I" PAGE ADDRESS REGISTERS
447
448      172340      KIPAR0= 172340
449      172342      KIPAR1= 172342
450      172344      KIPAR2= 172344
451      172346      KIPAR3= 172346
452      172350      KIPAR4= 172350
453      172352      KIPAR5= 172352
454      172354      KIPAR6= 172354
455      172356      KIPAR7= 172356
456
457      ;*KERNEL "D" PAGE ADDRESS REGISTERS
458
459      172360      KDPAR0= 172360
460      172362      KDPAR1= 172362
461      172364      KDPAR2= 172364
462      172366      KDPAR3= 172366
463      172370      KDPAR4= 172370
464      172372      KDPAR5= 172372
465      172374      KDPAR6= 172374
466      172376      KDPAR7= 172376
467
468      ;THESE ARE FLOATING POINT ACCUMULATOR EQUATES
469      000000      AC0= *0
470      000001      AC1= *1
471      000002      AC2= *2
472      000003      AC3= *3
473      000004      AC4= *4
474      000005      AC5= *5
475      000006      AC6= *6
476      000007      AC7= *7
477
478      000244      FPVEC= 244
479
480      ;THESE ARE CACHE REGISTER EQUATES
481      177746      CCR= 177746 ;CACHE CONTROL REGISTER
482      177744      MSER= 177744 ;MEMORY SYSTEM ERROR REGISTER
483      177752      HITMIS= 177752 ;HIT/MISS REGISTER
484      177766      CPereg= 177766 ;CPU ERROR REGISTER
485

```

```

486 ;MISCELLANEOUS DEFINITIONS
487 BEVENT= 177546 ;BEVENT CONTROL REGISTER
488 RCSR= 177560
489 RBUF= 177562
490 XCSR= 177564
491 XBUF= 177566
492 ERRTN= HALT
493 $TSTNU=1
494 ERRNUM= 1 ;INITIALIZE ERROR NUMBER COUNTER
495 ASWREG= 2000 ;SWR FOR APT--NO BEVENT TESTING
496
497
498 ;THIS EQUATE DEFINES THE BOTTOM OF THE PROGRAM STACK POINTER
499 STBOT= 1000
500
501 .ASECT
502 .SBTTL TRAP CATCHER
503
504 .*=0
505 ;*ALL UNUSED LOCATIONS OF THE VECTOR AREA CONTAIN
506 ;*A ".+2, IOT" SEQUENCE TO CATCH AND PROCESS ILLEGAL
507 ;*TRAPS AND INTERRUPTS THAT MIGHT OCCUR.
508 ;*THE IOT TRAP WHICH IS TAKEN ON THE ILLEGAL TRAP/INT
509 ;*TRAPS TO THE $SCOPE ROUTINE WHICH (IF THE RETURN PC IS
510 ;*LESS THAN 1002) JUMPS TO THE $ERROR ROUTINE.
511 ;*THE $ERROR ROUTINE WILL REPORT THE ERROR AS FOLLOWS:
512 ;* PC=YYYYYY UNEXPECTED TRAP TO XXX
513 ;*AND RETURN TO THE PROGRAM AT PC=YYYYYY+2
514 ;*WHERE XXX=LOCATION OF ILLEGAL TRAP
515 ;* YYYYYY=PC AT TIME OF TRAP
516 ;*NOTE: IF THE PROCESSOR IS NOT AN 11/05 THE PROGRAM
517 ;* CAN BE STARTED AT ADDRESS 0 AS WELL AS ADDRESS 200.
518 000000 000000 $4OCAT: HALT ;:HALT
519 000002 000737 BR -100 ;:BRANCH TO 177700 & TIME OUT (NOT ON
520 ;:11/05)
521 000004 001604 .WORD START ;:VECTOR TO STARTING ADDRESS
522 000006 000340 .WORD 340 ;:WITH PRIORITY LEVEL 7
523 .*=174
524 000174 000000 DISPREG: .WORD 0 ;:SOFTWARE DISPLAY REGISTER
525 000176 000000 SWREG: .WORD 0 ;:SOFTWARE SWITCH REGISTER
526
527 000200 000137 001604 .SBTTL STARTING ADDRESS)
528 JMP @START ;:GO TO START OF PROGRAM
529 .SBTTL ACT11 HOOKS
530
531 ;:*****
532 ;HOOKS REQUIRED BY ACT11
533 $SVPC= . ;SAVE PC
534 .+46
535 $ENDAD ;:1)SET LOC.46 TO ADDRESS OF $ENDAD IN , $EOP
536 .+52
537 .WORD 0 ;:2)SET LOC.52 TO ZERO
538 .+$SVPC ;: RESTORE PC
539 .SBTTL APT PARAMETER BLOCK
540
541 ;:*****
542 ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT

```

```

542 ;*****
543      .,$X,      ;;SAVE CURRENT LOCATION
544      .-24      ;;SET POWER FAIL TO POINT TO START OF PROGRAM
545 000024 000200 200      ;;FOR APT START UP
546      .-44      ;;POINT TO APT INDIRECT ADDRESS PNTR.
547 000044 000204 $APTHDR ;;POINT TO APT HEADER BLOCK
548      .-,$X      ;;RESET LOCATION COUNTER
549 ;*****
550 ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
551 ;INTERFACE SPEC.
552
553 000204 $APTHD:
554 000204 000000 $HIBTS: .WORD 0      ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
555 000206 001000 $MADR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
556 000210 000001 $TSTM: .WORD 1      ;;RUN TIM OF LONGEST TEST
557 000212 000002 $PASTM: .WORD 2     ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
558 000214 000000 $UNITM: .WORD 0     ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
559 000216 000014 .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX ETABLE(WORDS)
560      .-,$X      ;;SAVE CURRENT LOCATION COUNT
561      .-2
562 000002 0
563 000004 6
564 000006 4      ;SET UP SOME VECTORS
565      .-,$X      ;RESTORE LOCATION COUNT
566      .-1000

```

```

567 .SBTTL GLOBAL DATA SECTION
568
569 ***
570 ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
571 ; IN MORE THAN ONE TEST.
572 ;
573 .SBTTL APT MAILBOX ETABLE
574
575 ;*****
576 .EVEN
577 001000 $MAIL: ; APT MAILBOX
578 001000 000000 $MSGTY: .WORD AMSGTY ; MESSAGE TYPE CODE
579 001002 000000 $FATAL: .WORD AFATAL ; FATAL ERROR NUMBER
580 001004 000000 $TESTN: .WORD ATESTN ; TEST NUMBER
581 001006 000000 $PASS: .WORD APASS ; PASS COUNT
582 001010 000000 $DEVCT: .WORD ADEVCT ; DEVICE COUNT
583 001012 000000 $UNIT: .WORD AUNIT ; I/O UNIT NUMBER
584 001014 000000 $MSGAD: .WORD AMSGAD ; MESSAGE ADDRESS
585 001016 000000 $MSGLG: .WORD AMSGLG ; MESSAGE LENGTH
586 001020 $ETABLE: ; APT ENVIRONMENT TABLE
587 001020 000 $ENV: .BYTE AENV ; ENVIRONMENT BYTE
588 001021 000 $ENVM: .BYTE AENVM ; ENVIRONMENT MODE BITS
589 001022 002000 $SWREG: .WORD ASWREG ; APT SWITCH REGISTER
590 001024 000000 $USWR: .WORD AUSWR ; USER SWITCHES
591 001026 000000 $CPUOP: .WORD ACPUOP ; CPU TYPE, OPTIONS
592 ;* BITS 15-11-CPU TYPE
593 ;* 11/04-01,11/05-02,11/20-03,11/40-04,11/45-05
594 ;* 11/70-06,PDQ-07,Q-10
595 ;* BIT 10-REAL TIME CLOCK
596 ;* BIT 9-FLOATING POINT PROCESSOR
597 ;* BIT 8-MEMORY MANAGEMENT
598 001030 $ETEND:
599 .MEXIT
600
601 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE VECTOR DATA
602 ; WHEN THE TEST NEEDS TO HAVE AN ERROR CONDITION RESPOND DIFFERENTLY
603 ; FROM THE DEFAULT RESPONSE.
604 001030 000000 SLOC00: .WORD 0
605 001032 000000 SLOC01: .WORD 0
606
607 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE WORKING DATA.
608 001034 000000 EXPDAT: .WORD 0 ; STORES EXPECTED (GOOD) DATA FOR COMPARISONS
609 001036 000000 RECDAT: .WORD 0 ; STORES RECEIVED DATA TO BE VERIFIED
610 001040 000000 COUNT: .WORD 0 ; ERROR INDICATOR FOR FLOATING POINT TESTS
611 001042 000000 FLAG: .WORD 0 ; USED TO STORE "FLAG" CONDITIONS
612 001044 000000 ERRCNT: .WORD 0 ; STORAGE FOR ERROR COUNT
613 001046 177570 SWR: .WORD USWR ; STORAGE FOR SWITCH REGISTER ADDRESS
614 001050 177570 DISPLAY: .WORD DDISP ; STORAGE FOR DISPLAY REGISTER ADDRESS
615 001052 000000 $ERFLG: .WORD 0 ; ERROR FLAG
616 ; THESE LOCATIONS ARE USED BY MORE THAN ONE TEST AS LOOP COUNTERS
617 001054 000000 DCOUNT: .WORD 0
618 001056 000000 ALLCTR: .WORD 0
619 001060 000000 LOOPIN: .WORD 0
620 001062 000000 SAVSP1: .WORD 0 ; STORAGE FOR UNEXPECTED TRAP DATA
621 001064 000000 SAVSP2: .WORD 0
622

```

C2

623					
624	001066	000000	SAVSUP: .WORD	0	USED TO STORE SUPERVISOR STACK VALUE
625	001070	000000	SAVUSE: .WORD	0	USED TO STORE USER STACK VALUE
626	001072	000000	SAVMRO: .WORD	0	USED TO STORE MMU STATUS REGISTER 0 DATA
627	001074	000000	SAVMR1: .WORD	0	USED TO STORE MMU STATUS REGISTER 1 DATA
628	001076	000000	SAVMR2: .WORD	0	USED TO STORE MMU STATUS REGISTER 2 DATA
629	001100	000004	FLOAT: .BLKW	4	USED TO STORE VALUES FOR MMU TESTS
630	001110	000004	FLO: .BLKW	4	USED TO STORE VALUES FOR MMU TESTS

631
632
633
634
635
636
637

!!!!!!THIS IS IT. THE PROGRAM TEST LOCATION AND WRITE BUFFER!!!!!!!!!!!!!!!!!!!!!!
 TSTLOC: .BLKW 2

638
639 001120
640 001120 000002

```

641          .SBTTL  GLOBAL TEXT SECTION
642
643          ;**
644          ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
645          ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
646          ; MORE THAN ONE TEST.
647          ;**
648
649          ;
650          ; FORMAT STATEMENTS USED IN PRINT CALLS
651          ;
652
653 001124 005015 042523 020124 OPMSG2: .ASCIZ <CR><LF>/SET BIT 8 = 1 FOR 18 BIT SYSTEM/
654 001132 044502 020124 020070
655 001140 020075 020061 047506
656 001146 020122 034061 041040
657 001154 052111 051440 051531
658 001162 042524 000115
659 001166 005015 040503 044103 ERRMSG: .ASCIZ <CR><LF>/CACHE SYSTEM ERROR/
660 001174 020105 054523 052123
661 001202 046505 042440 051122
662 001210 051117 000
663 001213 015 042412 051122 MMUERR: .ASCIZ <CR><LF>/ERROR DURING MMU TESTING/
664 001220 051117 042040 051125
665 001226 047111 020107 046515
666 001234 020125 042524 052123
667 001242 047111 000107
668 001246 005015 051105 047522 ERR1: .ASCIZ <CR><LF>/ERROR # */
669 001254 020122 020043 000075
670 001262 005015 051105 047522 ERR2: .ASCIZ <CR><LF>/ERROR PC */
671 001270 020122 041520 036440
672 001276 000
673 001277 015 020012 020040 $CRLF: .ASCIZ <CR><LF>/ /
674 001304 000
675          001306          .EVEN

```


E2

GLOBAL AREAS
KDJ11A.MAC

MACY11 30AC1052)
20-MAR-84 11:19

20-MAR-84 11:31 PAGE 17

GLOBAL ERROR REPORT SECTION

SEQ 0017

076
077
078
079
080
081
082

.SBTTL GLOBAL ERROR REPORT SECTION

; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.

```

683      .SBTTL GLOBAL SUBROUTINES SECTION
684
685      ;**
686      ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
687      ; THAT ARE USED IN MORE THAN ONE TEST.
688      ;**
689      ;
690      ;MMU GLOBAL SUBROUTINES
691      ;
692      ;
693      ;ROUTINE TO INITIALIZE MEMORY MANAGEMENT
694      ;
695      MMU:  MOV     R0,-(SP)          ;SAVE CONTENTS OF REGISTERS
696           MOV     R1,-(SP)          ;
697           MOV     R2,-(SP)          ;
698           MOV     @177600,R0        ;
699           JSR    PC,PDR            ;INIT I AND D USER PDR'S
700           JSR    PC,PAR            ;INIT I USER PAR'S
701           JSR    PC,PAR            ;INIT D USER PAR'S
702           MOV     @172200,R0        ;
703           JSR    PC,PDR            ;INIT I AND D SUP PDR'S
704           JSR    PC,PAR            ;INIT I SUP PAR'S
705           JSR    PC,PAR            ;INIT D SUP PAR'S
706           JSR    PC,PDR            ;INIT I AND D KER PDR'S
707           JSR    PC,PAR            ;INIT I KER PAR'S
708           JSR    PC,PAR            ;INIT D KER PAR'S
709           MOV     @27,@172516       ;INIT MMR3
710           MOV     (SP)+,R2         ;RESTORE REGISTERS
711           MOV     (SP)+,R1         ;
712           MOV     (SP)+,R0         ;
713           RTS     PC               ;RETURN
714
715      ;ROUTINE TO INITIALIZE PDR'S
716      ;
717      PDR:  CLR     R2                ;INIT CNTR
718      PDR1: MOV     @77406,(R0)+      ;INIT PDR
719           ADD     @1,R2             ;INCREMENT CNTR
720           CMP     @16,,R2          ;ARE WE DONE?
721           BNE    PDR1             ;BRANCH IF NOT
722           RTS     PC               ;RETURN
723
724      ;ROUTINE TO INITIALIZE PAR'S
725      ;
726      PAR:  CLR     R1                ;SETUP TO INIT PAR
727      PAR1: MOV     R1,(R0)+         ;INIT PAR
728           ADD     @200,R1          ;GET READY FOR NEXT PAR
729           CMP     @1600,,R1        ;REACHED A PAR?
730           BNE    PAR1             ;BRANCH IF NOT
731           MOV     @177600,(R0)+    ;INIT PAR?
732           RTS     PC               ;RETURN
733
734      ;TIME OUT ROUTINE
735      ;
736      ADDTRP: INC     R5              ;INCREMENT TIME OUT FLAG
737           RTI                      ;RETURN
738

```

```

739 ;MMU TRAP ROUTINE
740 ;
741 001460 026727 177356 000001 MMUTRP: CMP FLAG,#1 ;ARE WE EXPECTING AN ABORT
742 001466 001403 BEQ 1$ ;YES GO ON
743 001470 104000 ERROR ;ALL ERRORS TO TRAP TO FMT VECTOR
744 001472 000001 .WORD 1 ;UNIQUE ERROR NUMBER
745 001474 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
746 001476 010046 1$: MOV R0, -(SP) ;SAVE CONTENTS OF REG 0
747 001500 013700 177776 MOV @#177776,R0 ;SAVE A COPY OF PSW
748 001504 072027 177764 ASH @-14,R0 ;LOOK AT BITS<15:14>
749 001510 020027 000002 CMP R0,#2 ;WAS PS<15:14>=10
750 001514 001001 BNE OK ;NO GO ON
751 001516 000411 BR NOTOK ;YES CHANGE BITS TO 00
752 001520 013700 177776 OK: MOV @#177776,R0 ;SAVE A COPY OF PSW
753 001524 072027 000002 ASH @2,R0 ;LOOK AT BITS<13:12>
754 001530 072027 177764 ASH @-14,R0 ;
755 001534 020027 000002 CMP R0,#2 ;WAS PS<13:12>=10
756 001540 001002 BNE OK1 ;NO GO ON
757 001542 005066 000004 NOTOK: CLR 4(SP) ;CLEAR ILLEGAL MODE FROM OLD PSW
758 001546 013767 177572 177316 OK1: MOV @#177572,SAVMR0 ;SAVE A COPY OF MMRO
759 001554 013767 177574 177312 MOV @#177574,SAVMR1 ;SAVE A COPY OF MMR1
760 001562 013767 177576 177306 MOV @#177576,SAVMR2 ;SAVE A COPY OF MMR2
761 001570 005037 177572 CLR @#177572 ;CLEAR ABORT BITS AND TURN MMU OFF
762 001574 005067 177242 CLR FLAG ;CLEAR MMU ABORT FLAG
763 001600 012600 MOV (SP)+,R0 ;RESTORE ORIGINAL CONTENTS OF REG 0
764 001602 000002 RTI ;RETURN

```

```

765 001604          START:
766 001604 012737 000014 177746   MOV    014,00CCR           ;SET CACHE TO FORCE MISS
767                                     .SBTTL INITIALIZE THE COMMON TAGS
768 001612 012706 001000          MOV    0STACK,SP         ;SETUP THE STACK POINTER
769                                     ;;INITIALIZE A FEW VECTORS
770 001616 012737 022304 000030   MOV    0$ERROR,00LMTVEC ;;EMT VECTOR FOR ERROR ROUTINE
771 001624 012737 000340 000032   MOV    0340,00ERRVEC+2 ;;LEVEL 7
772 001632 012737 021756 000034   MOV    0$TRAP,0TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
773 001640 012737 000340 000036   MOV    0340,00TRAPVEC+2;LEVEL 7
774 001646 005067 177134          CLR    $PASS             ;;CLEAR THE PASS COUNT
775 001652 016767 016116 016106   MOV    $ENDCT,$EOPCT    ;;SETUP END-OF-PROGRAM COUNTER
776 001660 105067 177166          CLRB   $ERFLG           ;;CLEAR THE ERPOR FLAG
777                                     ;;SIZE FOR A HARDWARE SWITCH REGISTER, IF NOT FOUND OR IT IS
778                                     ;;EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
779 001664 013746 000004          MOV    00ERRVEC,-(SP)   ;;SAVE ERROR VECTOR
780 001670 012737 001724 000004   MOV    064$,00ERRVEC   ;;SET UP ERROR VECTOR
781 001676 012767 177570 177142   MOV    0DSWR,SWR       ;;SETUP FOR A HARDWARE SWICH REGISTER
782 001704 012767 177570 177136   MOV    0DDISP,DISPLAY  ;;AND A HARDWARE DISPLAY REGISTER
783 001712 022777 177777 177126   CMP    0-1,0SWR        ;;TRY TO REFERENCE HARDWARE SWR
784 001720 001012          BNE    66$             ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
785                                     ;;AND THE HARDWARE SWR IS NOT = -1
786 001722 000403          BR     65$            ;;BRANCH IF NO TIMEOUT
787 001724 012716 001732          64$:  MOV    065$, (SP)    ;;SET UP FOR TRAP RETURN
788 001730 000002          RTI
789 001732 012767 000176 177106   65$:  MOV    0SWREG,SWR    ;;POINT TO SOFTWARE SWR
790 001740 012767 000174 177102   MOV    0DISPREG,DISPLAY
791 001746 012637 000004          66$:  MOV    (SP)+,00ERRVEC ;;RESTORE ERROR VECTOR
792
793 .MACRO  $$SETMAIL      ?$ARG1
794       CLR    $PASS     ;;CLEAR PASS COUNT
795       BITB   0APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
796       BEQ    $ARG1     ;;YES,USE NON-APT SWITCH
797       MOV    0$SWREG,SWR ;;NO,USE APT SWITCH REGISTER
798 $ARG1:
799 .ENDM  $$SETMAIL
800 001752 005067 177030          CLR    $PASS           ;;CLEAR PASS COUNT
801 001756 132767 000200 177035   BITB   0APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
802 001764 001403          BEQ    67$            ;;YES,USE NON-APT SWITCH
803 001766 012767 001022 177052   MOV    0$SWREG,SWR    ;;NO,USE APT SWITCH REGISTER
804 001774          67$:
805 001774 012737 022304 000020   MOV    0$ERROR,00IOTVEC ;;SET UP IOT VECTORS
806 002002 012737 000340 000022   MOV    0340,00IOTVEC+2 ;;TO GO TO ERROR ROUTINE
807 002010 005037 177766          CLR    00177766       ;;CLEAR CPU ERROR REGISTER
808 002014 012767 001460 176226   MOV    0MMUTRP,MMVEC
809 002022 104401 001124          TYPE   ,OPMSG2        ;OPERATOR MESSAGE 2
810 .SBTTL GET VALUE FOR SOFTWARE SWITCH REGISTER
811 002026 005737 000042          TST    0042           ;;ARE WE RUNNING UNDER XXDP/ACT?
812 002032 001012          BNE    68$           ;;BRANCH IF YES
813 002034 126727 176760 000001   CMPB   $ENV,01        ;;ARE WE RUNNING UNDER APT?
814 002042 001406          BEQ    68$           ;;BRANCH IF YES
815 002044 026727 176776 000176   CMP    SWP,0SWREG     ;;SOFTWARE SWITCH REG SELECTED?
816 002052 001005          BNE    69$           ;;BRANCH IF NO
817 002054 104406          GETSWR                ;;GET SOFT-SWR SETTINGS
818 002056 000403          BR     69$
819 002060 112767 000001 017666   68$:  MOVVB  01,$AUTOB      ;;SET AUTO-MODE INDICATOR
820 002066          69$:

```

```

821 002066 005067 176712 RESTART: CLR $TESTN ;RESET $TESTN TO ZERO
822 002072 012737 000014 177746 MOV #14,$CCR ;SET CACHE TO FORCE MISS
823
824 .SBTTL MEMORY MANAGEMENT TESTS
825 ;*****
826 ;*****
827 ; BEGIN MMU TESTING
828 ;*****
829 ;*****
830 002100 TSMMU1:
831 ;*****
832 ;+TEST 1 STATUS REGISTER TEST
833 ;*****
834 002100 TST1:
835 002100 005267 176700 INC $TESTN ;INCREMENT TEST NUMBER
836 002104 005067 175656 CLR CPEREG ;CLEAR CPU ERROR REGISTER
837 002110 005037 177572 CLR #0177572 ;TURN MMU OFF
838 002114 005037 001042 CLR #0FLAG ;CLEAR MMU TRAP FLAG
839 002120 013746 000004 MOV #04,-(SP) ;SAVE OLD VECTOR
840 002124 012737 001454 000004 MOV #ADDTRP,#04 ;SETUP NEW VECTOR
841 002132 005005 CLR R5 ;CLEAR FLAG
842 002134 013701 177572 MOV #0177572,R1 ;TEST MMR0
843 002140 013701 177574 MOV #0177574,R1 ;TEST MMR1
844 002144 013701 177576 MOV #0177576,R1 ;TEST MMR2
845 002150 013701 172516 MOV #0172516,R1 ;TEST MMR3
846 002154 012637 000004 MOV (SP)+,#04 ;RESTORE VECTOR
847 002160 020527 000000 CMP R5,#0 ;DID WE TRAP
848 002164 001403 BEQ 1$ ;NO, THEN BRANCH
849 002166 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
850 002170 000002 .WORD 2 ;UNIQUE ERROR NUMBER
851 002172 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
852
853 002174 1$: ;YES, GO TO ERROR
854
855 002174 TSMMU2:
856 ;*****
857 ;+TEST 2 ADDRESS TEST OF PARS,PDRS, AND FP REGS
858 ;*****
859 002174 TST2:
860 002174 005267 176604 INC $TESTN ;INCREMENT TEST NUMBER
861 002200 005067 175562 CLR CPEREG ;CLEAR CPU ERROR REGISTER
862 002204 005037 177572 CLR #0177572 ;MMU OFF
863 002210 005037 001042 CLR #0FLAG ;CLEAR MMU TRAP FLAG
864 002214 013746 000244 MOV #0244,-(SP) ;SAVE FP VECTOR
865 002220 013746 000246 MOV #0246,-(SP)
866 002224 013746 000004 MOV #04,-(SP) ;SAVE TIME OUT VECTOR
867 002230 012737 000246 000244 MOV #246,#0244 ;SETUP NEW FP VECTOR
868 002236 012737 000002 000246 MOV #2,#0246
869 002244 012737 001454 000004 MOV #ADDTRP,#04 ;SETUP NEW TIME OUT VECTOR
870 002252 005005 CLR R5 ;CLEAR TIMEOUT FLAG
871 002254 012700 172200 MOV #0172200,R0 ;LOAD ALL PARS AND PDRS WITH ZERO
872 002260 005020 1$: CLR (R0)+
873 002262 020027 172400 CMP R0,#0172400
874 002266 001374 BNE 1$
875 002270 012700 177600 MOV #0177600,R0
876 002274 005020 2$: CLR (R0)+

```

JP

877	002276	020027	177700		CMP	R0,#177700	:
878	002302	001374			BNE	2\$:
879	002304	170127	000200		LDFPS	#200	:
880	002310	012700	001100		MOV	#FLOAT,R0	;LOAD ACO-AC5 WITH 0
881	002314	005020			CLR	(R0)+	:
882	002316	005020			CLR	(R0)+	:
883	002320	005020			CLR	(R0)+	:
884	002322	005020			CLR	(R0)+	:
885	002324	012700	001100		MOV	#FLOAT,R0	:
886	002330	172410			LDD	(R0),ACO	:
887	002332	172510			LDD	(R0),AC1	:
888	002334	172610			LDD	(R0),AC2	:
889	002336	172710			LDD	(R0),AC3	:
890	002340	174004			STD	ACO,AC4	:
891	002342	174005			STD	ACO,AC5	:
892	002344	174500		3\$:	DIVD	ACO,AC1	;LOAD FEC WITH 4 AND FEA WITH #3\$
893	002346	170337	001110		STST	#FLO	;CHECK FEC FOR 4 AND FEA FOR #3\$
894	002352	012704	001110		MOV	#FLO,R4	:
895	002356	022427	000004		CMP	(R4)+,#4	:
896	002362	001403			BEQ	21\$:
897	002364	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
898	002366	000003			.WORD	3	;UNIQUE ERROR NUMBER
899	002370	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
900							:
901	002372	021427	002344	21\$:	CMP	(R4),#3\$:
902	002376	001403			BEQ	22\$:
903	002400	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
904	002402	000004			.WORD	4	;UNIQUE ERROR NUMBER
905	002404	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
906							:
907	002406	012704	172200	22\$:	MOV	#172200,R4	;CHECK EACH PAR, PDR FOR 0 THEN
908	002412	012701	000001		MOV	#1,R1	;WRITE A UNIQUE NUMBER TO IT
909	002416	010102		4\$:	MOV	R1,R2	:
910	002420	072227	000010		ASH	#10,R2	:
911	002424	021427	000000		CMP	(R4),#0	:
912	002430	001403			BEQ	5\$:
913	002432	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
914	002434	000005			.WORD	5	;UNIQUE ERROR NUMBER
915	002436	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
916							:
917	002440	010224		5\$:	MOV	R2,(R4)+	:
918	002442	005201			INC	R1	:
919	002444	020427	172400		CMP	R4,#172400	:
920	002450	001362			BNE	4\$:
921	002452	012704	177600		MOV	#177600,R4	:
922	002456	010102		6\$:	MOV	R1,R2	:
923	002460	072227	000010		ASH	#10,R2	:
924	002464	021427	000000		CMP	(R4),#0	:
925	002470	001403			BEQ	7\$:
926	002472	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
927	002474	000006			.WORD	6	;UNIQUE ERROR NUMBER
928	002476	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
929							:
930	002500	010224		7\$:	MOV	R2,(R4)+	:
931	002502	005201			INC	R1	:
932	002504	020427	177700		CMP	R4,#177700	:

```

933 002510 001362      BNE      6$
934 002512 012704 001110  MOV      #FLO,R4      ;CHECK AC5 FOR ALL ZEROES THEN LOAD A 6
935 002516 012703 001100  MOV      #FLOAT,R3
936 002522 174014      STD      ACO,(R4)
937 002524 172405      LDD      AC5,ACO
938 002526 174013      STD      ACO,(R3)
939 002530 012702 000004  MOV      #4,R2
940 002534 022327 000000 8$:  CMP      (R3)+,#0
941 002540 001403      BEQ      9$
942 002542 104000      ERROR
943 002544 000007      .WORD    7
944 002546 001213      .WORD    MMUERR      ;ALL ERRORS TO TRAP TO EMT VECTOR
945                                     ;UNIQUE ERROR NUMBER
946                                     ;ADDRESS OF ERROR MESSAGE
946 002550 005302      9$:  DEC      R2
947 002552 001370      BNE      8$
948 002554 012703 001100  MOV      #FLOAT,R3
949 002560 012713 000006  MOV      #6,(R3)
950 002564 172413      LDD      (R3),ACO
951 002566 174005      STD      ACO,AC5
952 002570 172404      LDD      AC4,ACO
953 002572 174013      STD      ACO,(R3)
954 002574 012702 000004  MOV      #4,R2
955 002600 022327 000000 10$:  CMP      (R3)+,#0
956 002604 001403      BEQ      11$
957 002606 104000      ERROR
958 002610 000010      .WORD    10
959 002612 001213      .WORD    MMUERR      ;ALL ERRORS TO TRAP TO EMT VECTOR
960                                     ;UNIQUE ERROR NUMBER
961                                     ;ADDRESS OF ERROR MESSAGE
961 002614 005302      11$:  DEC      R2
962 002616 001370      BNE      10$
963 002620 012703 001100  MOV      #FLOAT,R3
964 002624 012713 000005  MOV      #5,(R3)
965 002630 172413      LDD      (R3),ACO
966 002632 174004      STD      ACO,AC4
967 002634 012702 000004  MOV      #4,R2
968 002640 022427 000000 12$:  CMP      (R4)+,#0
969 002644 001403      BEQ      13$
970 002646 104000      ERROR
971 002650 000011      .WORD    11
972 002652 001213      .WORD    MMUERR      ;ALL ERRORS TO TRAP TO EMT VECTOR
973                                     ;UNIQUE ERROR NUMBER
974                                     ;ADDRESS OF ERROR MESSAGE
974 002654 005302      13$:  DEC      R2
975 002656 001370      BNE      12$
976 002660 012713 000001  MOV      #1,(R3)
977 002664 172413      LDD      (R3),ACO
978 002666 012704 001110  MOV      #FLO,R4
979 002672 012702 000004  MOV      #4,R2
980 002676 174114      STD      AC1,(R4)
981 002700 022427 000000 14$:  CMP      (R4)+,#0
982 002704 001403      BEQ      15$
983 002706 104000      ERROR
984 002710 000012      .WORD    12
985 002712 001213      .WORD    MMUERR      ;ALL ERRORS TO TRAP TO EMT VECTOR
986                                     ;UNIQUE ERROR NUMBER
987                                     ;ADDRESS OF ERROR MESSAGE
987 002714 005302      15$:  DEC      R2
988 002716 001370      BNE      14$

```

GLOBAL AREAS MACY11 30A(1052) 20-MAR-84 11:31 PAGE 24
 KDJ11A.MAC 20-MAR-84 11:19 T2 ADDRESS TEST OF PARS,PDRS, AND FP REGS

SEQ 0024

989	002720	012713	000002		MOV	02,(R3)	:	
990	002724	172513			LDD	(R3),AC1	:	
991	002726	012704	001110		MOV	0FLO,R4	:	CHECK AC2 FOR ALL ZEROES THEN LOAD A 3
992	002732	012702	000004		MOV	04,R2	:	
993	002736	174214			STD	AC2,(R4)	:	
994	002740	022427	000000	16\$:	CMP	(R4)+,00	:	
995	002744	001403			BEQ	17\$:	
996	002746	104000			ERROR		:	ALL ERRORS TO TRAP TO EMT VECTOR
997	002750	000013			.WORD	13	:	UNIQUE ERROR NUMBER
998	002752	001213			.WORD	MMUERR	:	ADDRESS OF ERROR MESSAGE
999							:	
1000	002754	005302		17\$:	DEC	R2	:	
1001	002756	001370			BNE	16\$:	
1002	002760	012713	000003		MOV	03,(R3)	:	
1003	002764	172613			LDD	(R3),AC2	:	
1004	002766	012704	001110		MOV	0FLO,R4	:	CHECK AC3 FOR ALL ZEROES THEN LOAD A 4
1005	002772	012702	000004		MOV	04,R2	:	
1006	002776	174314			STD	AC3,(R4)	:	
1007	003000	022427	000000	18\$:	CMP	(R4)+,00	:	
1008	003004	001403			BEQ	19\$:	
1009	003006	104000			ERROR		:	ALL ERRORS TO TRAP TO EMT VECTOR
1010	003010	000014			.WORD	14	:	UNIQUE ERROR NUMBER
1011	003012	001213			.WORD	MMUERR	:	ADDRESS OF ERROR MESSAGE
1012							:	
1013	003014	005302		19\$:	DEC	R2	:	
1014	003016	001370			BNE	18\$:	
1015	003020	012713	000004		MOV	04,(R3)	:	
1016	003024	172713			LDD	(R3),AC3	:	
1017	003026	012704	001110		MOV	0FLO,R4	:	CHECK FPS FOR 100204 THEN LOAD IT WITH 200
1018	003032	170214			STFPS	(R4)	:	
1019	003034	022714	100204		CMP	0100204,(R4)	:	
1020	003040	001403			BEQ	20\$:	
1021	003042	104000			ERROR		:	ALL ERRORS TO TRAP TO EMT VECTOR
1022	003044	000015			.WORD	15	:	UNIQUE ERROR NUMBER
1023	003046	001213			.WORD	MMUERR	:	ADDRESS OF ERROR MESSAGE
1024							:	
1025	003050	170127	000200	20\$:	LDFPS	0200	:	
1026	003054	012704	172200		MOV	0172200,R4	:	CHECK PDR, PAR FOR UNIQUE NUMBERS
1027	003060	012701	000001		MOV	01,R1	:	
1028	003064	010102		23\$:	MOV	R1,R2	:	
1029	003066	072227	000010		ASH	010,R2	:	
1030	003072	022402			CMP	(R4)+,R2	:	
1031	003074	001403			BEQ	24\$:	
1032	003076	104000			ERROR		:	ALL ERRORS TO TRAP TO EMT VECTOR
1033	003100	000016			.WORD	16	:	UNIQUE ERROR NUMBER
1034	003102	001213			.WORD	MMUERR	:	ADDRESS OF ERROR MESSAGE
1035							:	
1036	003104	005201		24\$:	INC	R1	:	
1037	003106	020427	172400		CMP	R4,0172400	:	
1038	003112	001364			BNE	23\$:	
1039	003114	012704	177600		MOV	0177600,R4	:	
1040	003120	010102		25\$:	MOV	R1,R2	:	
1041	003122	072227	000010		ASH	010,R2	:	
1042	003126	022402			CMP	(R4)+,R2	:	
1043	003130	001403			BEQ	26\$:	
1044	003132	104000			ERROR		:	ALL ERRORS TO TRAP TO EMT VECTOR

GLOBAL AREAS MACY11 30A(1052) 20-MAR-84 11:31 PAGE 25
 KDJ11A.MAC 20-MAR-84 11:19 T2 ADDRESS TEST OF PARS,PDRS, AND FP REGS

SEQ 0025

1045	003134	000017			.WORD	17			;UNIQUE ERROR NUMBER
1046	003136	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1047									
1048	003140	005201		26\$:	INC	R1			
1049	003142	020427	177700		CMP	R4,#177700			
1050	003146	001364			BNE	25\$			
1051	003150	012701	001100		MOV	#FLOAT,R1			;CHECK AC5 FOR #6
1052	003154	012704	001110		MOV	#FLO,R4			
1053	003160	174014			STD	AC0,(R4)			
1054	003162	172405			LDD	AC5,AC0			
1055	003164	174011			STD	AC0,(R1)			
1056	003166	022127	000006		CMP	(R1)+,#6			
1057	003172	001403			BEQ	27\$			
1058	003174	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1059	003176	000020			.WORD	20			;UNIQUE ERROR NUMBER
1060	003200	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1061									
1062	003202	012703	000003	27\$:	MOV	#3,R3			
1063	003206	022127	000000	28\$:	CMP	(R1)+,#0			
1064	003212	001403			BEQ	29\$			
1065	003214	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1066	003216	000021			.WORD	21			;UNIQUE ERROR NUMBER
1067	003220	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1068									
1069	003222	005303		29\$:	DEC	R3			
1070	003224	001370			BNE	28\$			
1071	003226	012701	001100		MOV	#FLOAT,R1			;CHECK AC4 FOR #5
1072	003232	172404			LDD	AC4,AC0			
1073	003234	174011			STD	AC0,(R1)			
1074	003236	022127	000005		CMP	(R1)+,#5			
1075	003242	001403			BEQ	30\$			
1076	003244	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1077	003246	000022			.WORD	22			;UNIQUE ERROR NUMBER
1078	003250	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1079									
1080	003252	012703	000003	30\$:	MOV	#3,R3			
1081	003256	022127	000000	31\$:	CMP	(R1)+,#0			
1082	003262	001403			BEQ	32\$			
1083	003264	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1084	003266	000023			.WORD	23			;UNIQUE ERROR NUMBER
1085	003270	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1086									
1087	003272	005303		32\$:	DEC	R3			
1088	003274	001370			BNE	31\$			
1089	003276	022427	000001		CMP	(R4)+,#1			;CHECK AC0 FOR #1
1090	003302	001403			BEQ	33\$			
1091	003304	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1092	003306	000024			.WORD	24			;UNIQUE ERROR NUMBER
1093	003310	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE
1094									
1095	003312	012703	000003	33\$:	MOV	#3,R3			
1096	003316	022427	000000	34\$:	CMP	(R4)+,#0			
1097	003322	001403			BEQ	35\$			
1098	003324	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
1099	003326	000025			.WORD	25			;UNIQUE ERROR NUMBER
1100	003330	001213			.WORD	MMUERR			;ADDRESS OF ERROR MESSAGE

```

1101
1102 003332 005303          35$: DEC      R3          ;
1103 003334 001370          BNE     34$          ;
1104 003336 012701 001100  MOV     @FLOAT,R1   ;CHECK AC1 FOR #2
1105 003342 174111          STD     AC1,(R1)    ;
1106 003344 022127 000002  CMP     (R1)+,#2    ;
1107 003350 001403          BEQ     36$          ;
1108 003352 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1109 003354 000026          .WORD  26          ;UNIQUE ERROR NUMBER
1110 003356 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1111
1112 003360 012703 000003  36$: MOV     #3,R3    ;
1113 003364 022127 000000  37$: CMP     (R1)+,#0 ;
1114 003370 001403          BEQ     38$          ;
1115 003372 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1116 003374 000027          .WORD  27          ;UNIQUE ERROR NUMBER
1117 003376 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1118
1119 003400 005303          38$: DEC      R3          ;
1120 003402 001370          BNE     37$          ;
1121 003404 012701 001100  MOV     @FLOAT,R1   ;CHECK AC2 FOR #3
1122 003410 174211          STD     AC2,(R1)    ;
1123 003412 022127 000003  CMP     (R1)+,#3    ;
1124 003416 001403          BEQ     39$          ;
1125 003420 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1126 003422 000030          .WORD  30          ;UNIQUE ERROR NUMBER
1127 003424 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1128
1129 003426 012703 000003  39$: MOV     #3,R3    ;
1130 003432 022127 000000  40$: CMP     (R1)+,#0 ;
1131 003436 001403          BEQ     41$          ;
1132 003440 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1133 003442 000031          .WORD  31          ;UNIQUE ERROR NUMBER
1134 003444 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1135
1136 003446 005303          41$: DEC      R3          ;
1137 003450 001370          BNE     40$          ;
1138 003452 012701 001100  MOV     @FLOAT,R1   ;CHECK AC3 FOR #4
1139 003456 174311          STD     AC3,(R1)    ;
1140 003460 022127 000004  CMP     (R1)+,#4    ;
1141 003464 001403          BEQ     42$          ;
1142 003466 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1143 003470 000032          .WORD  32          ;UNIQUE ERROR NUMBER
1144 003472 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1145
1146 003474 012703 000003  42$: MOV     #3,R3    ;
1147 003500 022127 000000  43$: CMP     (R1)+,#0 ;
1148 003504 001403          BEQ     44$          ;
1149 003506 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
1150 003510 000033          .WORD  33          ;UNIQUE ERROR NUMBER
1151 003512 001213          .WORD  MMUERR      ;ADDRESS OF ERROR MESSAGE
1152
1153 003514 005303          44$: DEC      R3          ;
1154 003516 001370          BNE     43$          ;
1155 003520 020527 000000  CMP     R5,#0       ;IS TIME OUT FLAG 0
1156 003524 001403          BEQ     45$          ;YES GO ON

```

```

1157 003526 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1158 003530 000034          .WORD      34          ;UNIQUE ERROR NUMBER
1159 003532 001213          .WORD      MMUERR      ;ADDRESS OF ERROR MESSAGE
1160                                     ;NO GO TO ERROR
1161 003534 012637 000004    45:  MOV      (SP)+,004    ;RESTORE TIME OUT VECTOR
1162 003540 012637 000246    MOV      (SP)+,00246    ;RESTORE FP VECTOR
1163 003544 012637 000244    MOV      (SP)+,00244    ;
1164
1165 003550          TSMMU3:
1166          ;*****
1167          ;TEST 3          WRITE ALL PARS/PDRS WITH ONES THEN ZERCS
1168          ;*****
1169          ;TEST3:
1170 003550          INC      $TESTN          ;INCREMENT TEST NUMBER
1171 003551 005037 177572    CLR      00177572      ;MMU OFF
1172 003560 005037 001040    CLR      00FLAG        ;CLEAR MMU ABORT FLAG
1173 003564 012703 172200    MOV      0172200,R3    ;LOAD ALL PARS AND PDRS WITH ONES
1174 003570 012723 177777    1:  MOV      0177777,(R3) ;
1175 003574 020327 172400    CMP      R3,0172400    ;
1176 003600 001373          BNE      1$           ;
1177 003602 012703 177600    MOV      0177600,R3    ;
1178 003606 012723 177777    2:  MOV      0177777,(R3) ;
1179 003612 020327 177700    CMP      R3,0177700    ;
1180 003616 001373          BNE      2$           ;
1181 003620 012703 172200    MOV      0172200,R3    ;CHECK SPDRS FOR ONES
1182 003624 022327 177416    3:  CMP      (R3)+,0177416 ;
1183 003630 001403          BEQ      4$           ;
1184 003632 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1185 003634 000035          .WORD      35          ;UNIQUE ERROR NUMBER
1186 003636 001213          .WORD      MMUERR      ;ADDRESS OF ERROR MESSAGE
1187
1188 003640 020327 172240    4:  CMP      R3,0172240    ;
1189 003644 001367          BNE      3$           ;
1190 003646 022327 177777    5:  CMP      (R3)+,0177777 ;
1191 003652 001403          BEQ      6$           ;
1192 003654 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1193 003656 000036          .WORD      36          ;UNIQUE ERROR NUMBER
1194 003660 001213          .WORD      MMUERR      ;ADDRESS OF ERROR MESSAGE
1195
1196 003662 020327 172300    6:  CMP      R3,0172300    ;
1197 003666 001367          BNE      5$           ;
1198 003670 022327 177416    7:  CMP      (R3)+,0177416 ;
1199 003674 001403          BEQ      8$           ;
1200 003676 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1201 003700 000037          .WORD      37          ;UNIQUE ERROR NUMBER
1202 003702 001213          .WORD      MMUERR      ;ADDRESS OF ERROR MESSAGE
1203
1204 003704 020327 172340    8:  CMP      R3,0172340    ;
1205 003710 001367          BNE      7$           ;
1206 003712 022327 177777    9:  CMP      (R3)+,0177777 ;
1207 003716 001403          BEQ      10$          ;
1208 003720 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1209 003722 000040          .WORD      40          ;UNIQUE ERROR NUMBER
1210 003724 001213          .WORD      MMUERR      ;ADDRESS OF ERROR MESSAGE
1211
1212 003726 020327 172400    10:  CMP      R3,0172400    ;

```

```

1213 003732 001367      BNE      9$
1214 003734 012703 177600      MOV      0177600,R3      ;CHECK UPDRS FOR ONES
1215 003740 022327 177416      11$:    CMP      (R3)+,0177416
1216 003744 001403      BEQ      12$
1217 003746 104000      ERROR
1218 003750 000041      .WORD   41              ;ALL ERRORS TO TRAP TO EMT VECTOR
1219 003752 001213      .WORD   MMUERR         ;UNIQUE ERROR NUMBER
1220                                     .WORD   ;ADDRESS OF ERROR MESSAGE
1221 003754 020327 177640      12$:    CMP      R3,0177640
1222 003760 001367      BNE      11$
1223 003762 022327 177777      13$:    CMP      (R3)+,0177777
1224 003766 001403      BEQ      14$
1225 003770 104000      ERROR
1226 003772 000042      .WORD   42              ;ALL ERRORS TO TRAP TO EMT VECTOR
1227 003774 001213      .WORD   MMUERR         ;UNIQUE ERROR NUMBER
1228                                     .WORD   ;ADDRESS OF ERROR MESSAGE
1229 003776 020327 177700      14$:    CMP      R3,0177700
1230 004002 001367      BNE      13$
1231 004004 012703 172200      MOV      0172200,R3      ;LOAD ALL PARS AND PDRS WITH ZEROS
1232 004010 012723 000000      15$:    MOV      00,(R3)+
1233 004014 020327 172400      CMP      R3,0172400
1234 004020 001373      BNE      15$
1235 004022 012703 177600      MOV      0177600,R3
1236 004026 012723 000000      16$:    MOV      00,(R3)+
1237 004032 020327 177700      CMP      R3,0177700
1238 004036 001373      BNE      16$
1239 004040 012703 172200      MOV      0172200,R3      ;CHECK ALL PARS AND PDRS FOR ZEROS
1240 004044 022327 000000      17$:    CMP      (R3)+,00
1241 004050 001403      BEQ      18$
1242 004052 104000      ERROR
1243 004054 000043      .WORD   43              ;ALL ERRORS TO TRAP TO EMT VECTOR
1244 004056 001213      .WORD   MMUERR         ;UNIQUE ERROR NUMBER
1245                                     .WORD   ;ADDRESS OF ERROR MESSAGE
1246 004060 020327 172400      18$:    CMP      R3,0172400
1247 004064 001367      BNE      17$
1248 004066 012703 177600      MOV      0177600,R3
1249 004072 022327 000000      19$:    CMP      (R3)+,00
1250 004076 001403      BEQ      20$
1251 004100 104000      ERROR
1252 004102 000044      .WORD   44              ;ALL ERRORS TO TRAP TO EMT VECTOR
1253 004104 001213      .WORD   MMUERR         ;UNIQUE ERROR NUMBER
1254                                     .WORD   ;ADDRESS OF ERROR MESSAGE
1255 004106 020327 177700      20$:    CMP      R3,0177700
1256 004112 001367      BNE      19$
1257
1258 004114      TSMMU4:
1259      ;*****
1260      ;*TEST 4      TEST FOR ADJACENT SHORTS IN PARS/PDRS
1261      ;*****
1262      TEST4:
1263 004114 005267 174664      INC      $TESTN         ;INCREMENT TEST NUMBER
1264 004120 005037 177572      CLR      00177572      ;MMU OFF
1265 004124 005067 174712      CLR      FLAG          ;CLEAR MMU ABORT FLAG
1266 004130 012700 172200      MOV      0172200,R0      ;LOAD SPDRS WITH ALTERNATING PATTERN
1267 004134 012720 052404      1$:    MOV      052404,(R0)+
1268 004140 012720 125012      MOV      0125012,(R0)+

```

1269	004144	020027	172240		CMP	R0,#172240	
1270	004150	001371			BNE	1#	
1271	004152	012720	125252	2#:	MOV	#125252,(R0),	;LOAD SPARS WITH ALTERNATING PATTERN
1272	004156	012720	052525		MOV	#52525,(R0),	
1273	004162	020027	172300		CMP	R0,#172300	
1274	004166	001371			BNE	2#	
1275	004170	012720	052404	3#:	MOV	#52404,(R0),	;LOAD KPDRS WITH ALTERNATING PATTERN
1276	004174	012720	125012		MOV	#125012,(R0),	
1277	004200	020027	172340		CMP	R0,#172340	
1278	004204	001371			BNE	3#	
1279	004206	012720	125252	4#:	MOV	#125252,(R0),	;LOAD KPARS WITH ALTERNATING PATTERN
1280	004212	012720	052525		MOV	#52525,(R0),	
1281	004216	020027	172400		CMP	R0,#172400	
1282	004222	001371			BNE	4#	
1283	004224	012700	177600		MOV	#177600,R0	;LOAD UPDRS WITH ALTERNATING PATTERN
1284	004230	012720	052404	5#:	MOV	#52404,(R0),	
1285	004234	012720	125012		MOV	#125012,(R0),	
1286	004240	020027	177640		CMP	R0,#177640	
1287	004244	001371			BNE	5#	
1288	004246	012720	125252	6#:	MOV	#125252,(R0),	;LOAD UPARS WITH ALTERNATING PATTERN
1289	004252	012720	052525		MOV	#52525,(R0),	
1290	004256	020027	177700		CMP	R0,#177700	
1291	004262	001371			BNE	6#	
1292							
1293	004264	012703	172200		MOV	#172200,R3	;CHECK SPDRS
1294	004270	022327	052404	7#:	CMP	(R3),#52404	
1295	004274	001403			BEQ	8#	
1296	004276	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1297	004300	000045			.WORD	45	;UNIQUE ERROR NUMBER
1298	004302	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1299							
1300	004304	022327	125012	8#:	CMP	(R3),#125012	
1301	004310	001403			BEQ	9#	
1302	004312	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1303	004314	000046			.WORD	46	;UNIQUE ERROR NUMBER
1304	004316	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1305							
1306	004320	020327	172240	9#:	CMP	R3,#172240	
1307	004324	001361			BNE	7#	
1308	004326	022327	125252	10#:	CMP	(R3),#125252	;CHECK SPARS
1309	004332	001403			BEQ	11#	
1310	004334	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1311	004336	000047			.WORD	47	;UNIQUE ERROR NUMBER
1312	004340	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1313							
1314	004342	022327	052525	11#:	CMP	(R3),#52525	
1315	004346	001403			BEQ	12#	
1316	004350	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1317	004352	000050			.WORD	50	;UNIQUE ERROR NUMBER
1318	004354	001213			.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1319							
1320	004356	020327	172300	12#:	CMP	R3,#172300	
1321	004362	001361			BNE	10#	
1322	004364	022327	052404	13#:	CMP	(R3),#52404	;CHECK KPDRS
1323	004370	001403			BEQ	14#	
1324	004372	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR

```

1325 004374 000051 .WORD 51 ;UNIQUE ERROR NUMBER
1326 004376 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1327
1328 004400 022327 125012 14$: CMP (R3)+,0125012 ;
1329 004404 001403 BEQ 15$ ;
1330 004406 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1331 004410 000052 .WORD 52 ;UNIQUE ERROR NUMBER
1332 004412 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1333
1334 004414 020327 172340 15$: CMP R3,0172340 ;
1335 004420 001361 BNE 13$ ;
1336 004422 022327 125252 16$: CMP (R3)+,0125252 ;CHECK KPARS
1337 004426 001403 BEQ 17$ ;
1338 004430 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1339 004432 000053 .WORD 53 ;UNIQUE ERROR NUMBER
1340 004434 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1341
1342 004436 022327 052525 17$: CMP (R3)+,052525 ;
1343 004442 001403 BEQ 18$ ;
1344 004444 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1345 004446 000054 .WORD 54 ;UNIQUE ERROR NUMBER
1346 004450 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1347
1348 004452 020327 172400 18$: CMP R3,0172400 ;
1349 004456 001361 BNE 16$ ;
1350 004460 012703 177600 MOV 0177600,R3 ;CHECK UPDRS
1351 004464 022327 052404 19$: CMP (R3)+,052404 ;
1352 004470 001403 BEQ 20$ ;
1353 004472 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1354 004474 000055 .WORD 55 ;UNIQUE ERROR NUMBER
1355 004476 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1356
1357 004500 022327 125012 20$: CMP (R3)+,0125012 ;
1358 004504 001403 BEQ 21$ ;
1359 004506 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1360 004510 000056 .WORD 56 ;UNIQUE ERROR NUMBER
1361 004512 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1362
1363 004514 020327 177640 21$: CMP R3,0177640 ;
1364 004520 001361 BNE 19$ ;
1365 004522 022327 125252 22$: CMP (R3)+,0125252 ;CHECK UPARS
1366 004526 001403 BEQ 23$ ;
1367 004530 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1368 004532 000057 .WORD 57 ;UNIQUE ERROR NUMBER
1369 004534 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1370
1371 004536 022327 052525 23$: CMP (R3)+,052525 ;
1372 004542 001403 BEQ 24$ ;
1373 004544 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1374 004546 000060 .WORD 60 ;UNIQUE ERROR NUMBER
1375 004550 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1376
1377 004552 020327 177700 24$: CMP R3,0177700 ;
1378 004556 001361 BNE 22$ ;
1379
1380 ;REVERSE ALTERNATING PATTERN

```

```

1381 ;
1382 004560 012700 172200 ; MOV #172200,R0 ;LOAD SPDRS WITH REVERSE PATTERN
1383 004564 012720 125012 25$: MOV #125012,(R0);
1384 004570 012720 052404 MOV #52404,(R0);
1385 004574 020027 172240 CMP R0,#172240 ;
1386 004600 001371 BNE 25$ ;
1387 004602 012720 052525 26$: MOV #52525,(R0); ;LOAD SPARS WITH REVERSE PATTERN
1388 004606 012720 125252 MOV #125252,(R0);
1389 004612 020027 172300 CMP R0,#172300 ;
1390 004616 001371 BNE 26$ ;
1391 004620 012720 125012 27$: MOV #125012,(R0); ;LOAD KPDRS WITH REVERSE PATTERN
1392 004624 012720 052404 MOV #52404,(R0);
1393 004630 020027 172340 CMP R0,#172340 ;
1394 004634 001371 BNE 27$ ;
1395 004636 012720 052525 28$: MOV #52525,(R0); ;LOAD KPARS WITH REVERSE PATTERN
1396 004642 012720 125252 MOV #125252,(R0);
1397 004646 020027 172400 CMP R0,#172400 ;
1398 004652 001371 BNE 28$ ;
1399 004654 012700 177600 MOV #177600,R0 ;LOAD UPDRS WITH REVERSE PATTERN
1400 004660 012720 125012 29$: MOV #125012,(R0);
1401 004664 012720 052404 MOV #52404,(R0);
1402 004670 020027 177640 CMP R0,#177640 ;
1403 004674 001371 BNE 29$ ;
1404 004676 012720 052525 30$: MOV #52525,(R0); ;LOAD UPARS WITH REVERSE PATTERN
1405 004702 012720 125252 MOV #125252,(R0);
1406 004706 020027 177700 CMP R0,#177700 ;
1407 004712 001371 BNE 30$ ;
1408 ;
1409 004714 012703 172200 ; MOV #172200,R3 ;CHECK SPDRS
1410 004720 022327 125012 31$: CMP (R3),#125012 ;
1411 004724 001403 BEQ 32$ ;
1412 004726 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1413 004730 000061 .WORD 61 ;UNIQUE ERROR NUMBER
1414 004732 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1415 ;
1416 004734 022327 052404 32$: CMP (R3),#52404 ;
1417 004740 001403 BEQ 33$ ;
1418 004742 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1419 004744 000062 .WORD 62 ;UNIQUE ERROR NUMBER
1420 004746 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1421 ;
1422 004750 020327 172240 33$: CMP R3,#172240 ;
1423 004754 001361 BNE 31$ ;
1424 004756 022327 052525 34$: CMP (R3),#52525 ;CHECK SPARS
1425 004762 001403 BEQ 35$ ;
1426 004764 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1427 004766 000063 .WORD 63 ;UNIQUE ERROR NUMBER
1428 004770 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1429 ;
1430 004772 022327 125252 35$: CMP (R3),#125252 ;
1431 004776 001403 BEQ 36$ ;
1432 005000 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1433 005002 000064 .WORD 64 ;UNIQUE ERROR NUMBER
1434 005004 001213 .WORD MMUERR ;ADDRESS OF ERROR MESSAGE
1435 ;
1436 005006 020327 172300 36$: CMP R3,#172300 ;

```

1437	005012	001361			BNE	34\$		
1438	005014	022327	125012	37\$:	CMP	(R3)+, #125012		; CHECK KPDRS
1439	005020	001403			BEQ	38\$		
1440	005022	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1441	005024	000065			.WORD	65		; UNIQUE ERROR NUMBER
1442	005026	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1443								
1444	005030	022327	052404	38\$:	CMP	(R3)+, #52404		
1445	005034	001403			BEQ	39\$		
1446	005036	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1447	005040	000066			.WORD	66		; UNIQUE ERROR NUMBER
1448	005042	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1449								
1450	005044	020327	172340	39\$:	CMP	R3, #172340		
1451	005050	001361			BNE	37\$		
1452	005052	022327	052525	40\$:	CMP	(R3)+, #52525		; CHECK KPARS
1453	005056	001403			BEQ	41\$		
1454	005060	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1455	005062	000067			.WORD	67		; UNIQUE ERROR NUMBER
1456	005064	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1457								
1458	005066	022327	125252	41\$:	CMP	(R3)+, #125252		
1459	005072	001403			BEQ	42\$		
1460	005074	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1461	005076	000070			.WORD	70		; UNIQUE ERROR NUMBER
1462	005100	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1463								
1464	005102	020327	172400	42\$:	CMP	R3, #172400		
1465	005106	001361			BNE	40\$		
1466	005110	012703	177600		MOV	#177600, R3		; CHECK UPDRS
1467	005114	022327	125012	43\$:	CMP	(R3)+, #125012		
1468	005120	001403			BEQ	44\$		
1469	005122	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1470	005124	000071			.WORD	71		; UNIQUE ERROR NUMBER
1471	005126	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1472								
1473	005130	022327	052404	44\$:	CMP	(R3)+, #52404		
1474	005134	001403			BEQ	45\$		
1475	005136	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1476	005140	000072			.WORD	72		; UNIQUE ERROR NUMBER
1477	005142	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1478								
1479	005144	020327	177640	45\$:	CMP	R3, #177640		
1480	005150	001361			BNE	43\$		
1481	005152	022327	052525	46\$:	CMP	(R3)+, #52525		; CHECK UPARS
1482	005156	001403			BEQ	47\$		
1483	005160	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1484	005162	000073			.WORD	73		; UNIQUE ERROR NUMBER
1485	005164	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1486								
1487	005166	022327	125252	47\$:	CMP	(R3)+, #125252		
1488	005172	001403			BEQ	48\$		
1489	005174	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1490	005176	000074			.WORD	74		; UNIQUE ERROR NUMBER
1491	005200	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1492								


```

1493 005202 020527 177700 48$: CMP R3,#177700 ;
1494 005206 001361 BNE 46$ ;
1495
1496 005210 TSMMU5:
1497 ;*****
1498 ;*TEST 5 TEST MMRO ABORT BITS
1499 ;*****
1500 005210 TST5:
1501 005210 005267 173570 _NC $TESTN ; INCREMENT TEST NUMBER
1502 005214 012737 160000 177572 MOV #150000,#177572 ; LOAD MMRO<15:13>=111
1503 005222 005067 173614 CLR FLAG ; CLEAR MMU ABORT FLAG
1504 005226 013700 177572 MOV #SRO,R0 ; SAVE SRO IN RO
1505 005232 042700 000176 BIC #176,R0 ; CLEAR UNDEFINED BITS FROM SRO
1506 005236 020027 160000 CMP R0,#160000 ; CHECK MMRO
1507 005242 001403 BEQ 1$ ;
1508 005244 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
1509 005246 000075 .WORD 75 ; UNIQUE ERROR NUMBER
1510 005250 001213 .WORD MMUERR ; ADDRESS OF ERROR MESSAGE
1511
1512 005252 005037 177572 1$: CLR #177572 ; LOAD MMRO=0
1513 005256 013700 177572 MOV #SRO,R0 ; SAVE SRO IN RO
1514 005262 042700 000176 BIC #176,R0 ; CLEAR UNDEFINED BITS FROM SRO
1515 005266 020027 000000 CMP R0,#0 ; CHECK MMRO
1516 005272 001403 BEQ 2$ ;
1517 005274 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
1518 005276 000076 .WORD 76 ; UNIQUE ERROR NUMBER
1519 005300 001213 .WORD MMUERR ; ADDRESS OF ERROR MESSAGE
1520
1521 005302 012737 120000 177572 2$: MOV #120000,#177572 ; LOAD MMRO<15:13>=101
1522 005310 013700 177572 MOV #SRO,R0 ; SAVE SRO IN RO
1523 005314 042700 000176 BIC #176,R0 ; CLEAR UNDEFINED BITS FROM SRO
1524 005320 020027 120000 CMP R0,#120000 ; CHECK MMRO
1525 005324 001403 BEQ 3$ ;
1526 005326 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
1527 005330 000077 .WORD 77 ; UNIQUE ERROR NUMBER
1528 005332 001213 .WORD MMUERR ; ADDRESS OF ERROR MESSAGE
1529
1530 005334 012737 040000 177572 3$: MOV #40000,#177572 ; LOAD MMRO<15:13>=010
1531 005342 013700 177572 MOV #SRO,R0 ; SAVE SRO IN RO
1532 005346 042700 000176 BIC #176,R0 ; CLEAR UNDEFINED BITS FROM SRO
1533 005352 020027 040000 CMP R0,#40000 ; CHECK MMRO
1534 005356 001403 BEQ 4$ ;
1535 005360 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
1536 005362 000100 .WORD 100 ; UNIQUE ERROR NUMBER
1537 005364 001213 .WORD MMUERR ; ADDRESS OF ERROR MESSAGE
1538 005366 4$:
1539
1540 005366 TSMMU6:
1541 ;*****
1542 ;*TEST 6 TEST MMR3 BITS 5-0
1543 ;*****
1544 005366 TST6:
1545 005366 005267 173412 INC $TESTN ; INCREMENT TEST NUMBER
1546 005372 005037 177572 CLR #177572 ; MMU OFF
1547 005376 005067 173440 CLR FLAG ; CLEAR MMU ABORT FLAG
1548 005402 012737 000077 172516 MOV #77,#172516 ; LOAD MMR3<5:0>=77

```


1605	005670	012737	140000	177776		MOV	#140000,#177776	;POINT TO USER SPACE
1606	005676	020637	001070			CMP	R6,#0SAVUSE	;IS USER SP CORRECT
1607	005702	001403				BEQ	100\$;YES GO ON
1608	005704	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1609	005706	000106				.WORD	106	;UNIQUE ERROR NUMBER
1610	005710	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1611								;NO GO TO ERROR
1612	005712	012737	040000	177776	100\$:	MOV	#40000,#177776	;POINT TO SUPERVISOR SPACE
1613	005720	020637	001066			CMP	R6,#0SAVSUP	;IS SUPERVISOR SP CORRECT
1614	005724	001403				BEQ	200\$;YES GO ON
1615	005726	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1616	005730	000107				.WORD	107	;UNIQUE ERROR NUMBER
1617	005732	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1618								;NO GO TO ERROR
1619	005734	023727	000244	135072	200\$:	CMP	#0244,#135072	;IS TEST DATA OK
1620	005742	001403				BEQ	2\$;YES GO ON
1621	005744	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1622	005746	000110				.WORD	110	;UNIQUE ERROR NUMBER
1623	005750	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1624								;NO GO TO ERROR
1625	005752	020327	000246		2\$:	CMP	R3,#246	;IS R3 CORRECT
1626	005756	001403				BEQ	3\$;YES GO ON
1627	005760	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1628	005762	000111				.WORD	111	;UNIQUE ERROR NUMBER
1629	005764	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1630								;NO GO TO ERROR
1631	005766	005037	177776		3\$:	CLR	#177776	;SET PSW TO KERNEL MODE
1632	005772	022627	135072			CMP	(SP),#135072	;IS KERNEL STACK CORRECT
1633	005776	001403				BEQ	4\$;YES GO ON
1634	006000	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1635	006002	000112				.WORD	112	;UNIQUE ERROR NUMBER
1636	006004	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1637								;NO GO TO ERROR
1638	006006	021627	177777		4\$:	CMP	(SP),#177777	;IS STACK CORRECT
1639	006012	001403				BEQ	5\$;YES GO ON
1640	006014	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1641	006016	000113				.WORD	113	;UNIQUE ERROR NUMBER
1642	006020	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1643								;NO GO TO ERROR
1644	006022	012737	030017	177776	5\$:	MOV	#30017,#177776	;SETUP PSW
1645	006030	012737	173621	000244		MOV	#173621,#244	;SETUP TEST LOCATION
1646	006036	012701	000244			MOV	#244,R1	;SETUP R1
1647	006042	005237	177572			INC	#177572	;TURN MMU ON
1648	006046	006511				MFPI	(R1)	;TEST INSTRUCTION
1649	006050	022737	030011	177776		CMP	#30011,#177776	;IS PSW CORRECT
1650	006056	001403				BEQ	300\$;YES GO ON
1651	006060	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1652	006062	000114				.WORD	114	;UNIQUE ERROR NUMBER
1653	006064	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE
1654								;NO GO TO ERROR
1655	006066	005037	177572		300\$:	CLR	#177572	;TURN MMU OFF
1656	006072	023727	000244	173621		CMP	#244,#173621	;IS TEST LOCATION CORRECT
1657	006100	001403				BEQ	301\$;YES GO ON
1658	006102	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1659	006104	000115				.WORD	115	;UNIQUE ERROR NUMBER
1660	006106	001213				.WORD	MMUERR	;ADDRESS OF ERROR MESSAGE

```

1661
1662 006110 020127 000244 301$: CMP R1,#244 ;NO GO TO ERROR
1663 006114 001403 BEQ 302$ ;IS R1 CORRECT
1664 006116 104000 ERROR ;YES GO ON
1665 006120 000116 .WORD 116 ;ALL ERRORS TO TRAP TO EMT VECTOR
1666 006122 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1667 ;ADDRESS OF ERROR MESSAGE
1668 006124 005037 177776 302$: CLR @#177776 ;NO GO TO ERROR
1669 006130 022627 173621 CMP (SP)+,#173621 ;SET PSW TO KERNEL MODE
1670 006134 001403 BEQ 303$ ;IS STACK CORRECT
1671 006136 104000 ERROR ;YES GO ON
1672 006140 000117 .WORD 117 ;ALL ERRORS TO TRAP TO EMT VECTOR
1673 006142 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1674 ;ADDRESS OF ERROR MESSAGE
1675 006144 021627 177777 303$: CMP (SP),#177777 ;NO GO TO ERROR
1676 006150 001403 BEQ 304$ ;IS STACK CORRECT
1677 006152 104000 ERROR ;YES GO ON
1678 006154 000120 .WORD 120 ;ALL ERRORS TO TRAP TO EMT VECTOR
1679 006156 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1680 ;ADDRESS OF ERROR MESSAGE
1681 006160 005003 304$: CLR R3 ;NO GO TO ERROR
1682 006162 005237 177572 INC @#177572 ;SETUP SOURCE FOR NEXT TEST
1683 006166 006503 MFPI R3 ;TURN MMU ON
1684 006170 022737 000004 177776 CMP #4,@#177776 ;TEST INSTRUCTION
1685 006176 001403 BEQ 6$ ;IS PSW CORRECT
1686 006200 104000 ERROR ;YES GO ON
1687 006202 000121 .WORD 121 ;ALL ERRORS TO TRAP TO EMT VECTOR
1688 006204 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1689 ;ADDRESS OF ERROR MESSAGE
1690 006206 005037 177572 6$: CLR @#177572 ;NO GO TO ERROR
1691 006212 020327 000000 CMP R3,#0 ;TURN MMU OFF
1692 006216 001403 BEQ 7$ ;IS R3 CORRECT
1693 006220 104000 ERROR ;YES GO ON
1694 006222 000122 .WORD 122 ;ALL ERRORS TO TRAP TO EMT VECTOR
1695 006224 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1696 ;ADDRESS OF ERROR MESSAGE
1697 006226 022627 000000 7$: CMP (SP)+,#0 ;NO GO TO ERROR
1698 006232 001403 BEQ 8$ ;IS STACK CORRECT
1699 006234 104000 ERROR ;YES GO ON
1700 006236 000123 .WORD 123 ;ALL ERRORS TO TRAP TO EMT VECTOR
1701 006240 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1702 ;ADDRESS OF ERROR MESSAGE
1703 006242 022627 177777 8$: CMP (SP)+,#177777 ;NO GO TO ERROR
1704 006246 001403 BEQ 9$ ;IS STACK CORRECT
1705 006250 104000 ERROR ;YES GO ON
1706 006252 000124 .WORD 124 ;ALL ERRORS TO TRAP TO EMT VECTOR
1707 006254 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1708 ;ADDRESS OF ERROR MESSAGE
1709 006256 012637 000244 9$: MOV (SP)+,@#244 ;NO GO TO ERROR
1710 ;RESTORE TEST LOCATION
1711 ;
1712 006262 ;TSMM6B;
1713 ;*****
1714 ;*TEST 10 TEST MFPI (MOVE FROM PREVIOUS DATA SPACE)
1715 ;*****
1716 006262 ;TST10;

```

1717	006262	005267	172516		INC	\$TESTN		; INCREMENT TEST NUMBER
1718	006266	005037	177572		CLR	00177572		; MMU OFF
1719	006272	005037	001042		CLR	00FLAG		; CLEAR MMU ABORT FLAG
1720	006276	012737	140000	177776	MOV	0140000,00177776		; POINT TO USER SPACE
1721	006304	010637	001070		MOV	R6,00SAVUSE		; SAVE USER SP
1722	006310	012737	040000	177776	MOV	040000,00177776		; POINT TO SUPERVISOR SPACE
1723	006316	010637	001066		MOV	R6,00SAVSUP		; SAVE SUPERVISOR SP
1724	006322	012737	030000	177776	MOV	030000,00177776		; SETUP PSW
1725	006330	004767	172752		JSR	PC,MMU		; INIT MMU
1726	006334	012737	000027	172516	MOV	027,00172516		; SETUP MMR3
1727	006342	013746	000244		MOV	00244,-(SP)		; SAVE DATA AT TEST LOCATION
1728	006346	012746	177777		MOV	0177777,-(SP)		; PUT KNOWN DATA ON TOP OF STACK
1729	006352	012737	157002	000244	MOV	0157002,00244		; SETUP DATA AT TEST LOCATION
1730	006360	012767	077400	171212	MOV	077400,UIPDRO		; SETUP UIPDRO TO ABORT
1731	006366	012703	000244		MOV	0244,R3		; SETUP POINTER TO TEST LOCATION
1732	006372	005237	177572		INC	00177572		; TURN MMU ON
1733	006376	106523			MFPD	(R3),		; TEST INSTRUCTION
1734	006400	022737	030010	177776	CMP	030010,00177776		; IS PSW CORRECT
1735	006406	001403			BEQ	1000		; YES GO ON
1736	006410	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1737	006412	000125			.WORD	125		; UNIQUE ERROR NUMBER
1738	006414	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1739								; NO GO TO ERROR
1740	006416	005037	177572	1000	CLR	00177572		; TURN MMU OFF
1741	006422	012737	140000	177776	MOV	0140000,00177776		; POINT TO USER SPACE
1742	006430	020637	001070		CMP	R6,00SAVUSE		; IS USER SP CORRECT
1743	006434	001403			BEQ	1000		; YES GO ON
1744	006436	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1745	006440	000126			.WORD	126		; UNIQUE ERROR NUMBER
1746	006442	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1747								; NO GO TO ERROR
1748	006444	012737	040000	177776	MOV	040000,00177776		; POINT TO SUPERVISOR SPACE
1749	006452	020637	001066		CMP	R6,00SAVSUP		; IS SUPERVISOR SP CORRECT
1750	006456	001403			BEQ	2000		; YES GO ON
1751	006460	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1752	006462	000127			.WORD	127		; UNIQUE ERROR NUMBER
1753	006464	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1754								; NO GO TO ERROR
1755	006466	023727	000244	157002	CMP	00244,0157002		; IS TEST DATA OK
1756	006474	001403			BEQ	2000		; YES GO ON
1757	006476	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1758	006500	000130			.WORD	130		; UNIQUE ERROR NUMBER
1759	006502	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1760								; NO GO TO ERROR
1761	006504	020327	000246	2000	CMP	R3,0246		; IS R3 CORRECT
1762	006510	001403			BEQ	3000		; YES GO ON
1763	006512	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1764	006514	000131			.WORD	131		; UNIQUE ERROR NUMBER
1765	006516	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE
1766								; NO GO TO ERROR
1767	006520	005037	177776	3000	CLR	00177776		; SET PSW TO KERNEL MODE
1768	006524	022627	157002		CMP	(SP),0157002		; IS KERNEL STACK CORRECT
1769	006530	001403			BEQ	4000		; YES GO ON
1770	006532	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1771	006534	000132			.WORD	132		; UNIQUE ERROR NUMBER
1772	006536	001213			.WORD	MMUERR		; ADDRESS OF ERROR MESSAGE

```

1773
1774 006540 021627 177777 4$: CMP (SP),#177777 ;NO GO TO ERROR
1775 006544 001403 BEQ 5$ ;IS STACK CORRECT
1776 006546 104000 ERROR ;YES GO ON
1777 006550 000133 .WORD 133 ;ALL ERRORS TO TRAP TO EMT VECTOR
1778 006552 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1779 ;ADDRESS OF ERROR MESSAGE
1780 006554 012737 030017 177776 5$: MOV #30017,#177776 ;NO GO TO ERROR
1781 006562 012737 103456 000244 MOV #103456,#244 ;SETUP PSW
1782 006570 012701 000244 MOV #244,R1 ;SETUP TEST LOCATION
1783 006574 005237 177572 INC #177572 ;SETUP R1
1784 006600 106511 MFPD (R1) ;TURN MMU ON
1785 006602 022737 030011 177776 CMP #30011,#177776 ;TEST INSTRUCTION
1786 006610 001403 BEQ 300$ ;IS PSW CORRECT
1787 006612 104000 ERROR ;YES GO ON
1788 006614 000134 .WORD 134 ;ALL ERRORS TO TRAP TO EMT VECTOR
1789 006616 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1790 ;ADDRESS OF ERROR MESSAGE
1791 006620 005037 177572 300$: CLR #177572 ;NO GO TO ERROR
1792 006624 023727 000244 103456 CMP #244,#103456 ;TURN MMU OFF
1793 006632 001403 BEQ 301$ ;IS TEST LOCATION CORRECT
1794 006634 104000 ERROR ;YES GO ON
1795 006636 000135 .WORD 135 ;ALL ERRORS TO TRAP TO EMT VECTOR
1796 006640 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1797 ;ADDRESS OF ERROR MESSAGE
1798 006642 020127 000244 301$: CMP R1,#244 ;NO GO TO ERROR
1799 006646 001403 BEQ 302$ ;IS R1 CORRECT
1800 006650 104000 ERROR ;YES GO ON
1801 006652 000136 .WORD 136 ;ALL ERRORS TO TRAP TO EMT VECTOR
1802 006654 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1803 ;ADDRESS OF ERROR MESSAGE
1804 006656 005037 177776 302$: CLR #177776 ;NO GO TO ERROR
1805 006662 022627 103456 CMP (SP)+,#103456 ;SET PSW TO KERNEL MODE
1806 006666 001403 BEQ 303$ ;IS STACK CORRECT
1807 006670 104000 ERROR ;YES GO ON
1808 006672 000137 .WORD 137 ;ALL ERRORS TO TRAP TO EMT VECTOR
1809 006674 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1810 ;ADDRESS OF ERROR MESSAGE
1811 006676 021627 177777 303$: CMP (SP),#177777 ;NO GO TO ERROR
1812 006702 001403 BEQ 304$ ;IS STACK CORRECT
1813 006704 104000 ERROR ;YES GO ON
1814 006706 000140 .WORD 140 ;ALL ERRORS TO TRAP TO EMT VECTOR
1815 006710 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1816 ;ADDRESS OF ERROR MESSAGE
1817 006712 012737 030017 177776 304$: MOV #30017,#177776 ;NO GO TO ERROR
1818 006720 012737 113672 000244 MOV #113672,#244 ;SETUP PSW
1819 006726 012701 000246 MOV #246,R1 ;SETUP TEST LOCATION
1820 006732 005237 177572 INC #177572 ;SETUP R1
1821 006736 106541 MFPD -(R1) ;TURN MMU ON
1822 006740 022737 030011 177776 CMP #30011,#177776 ;TEST INSTRUCTION
1823 006746 001403 BEQ 400$ ;IS PSW CORRECT
1824 006750 104000 ERROR ;YES GO ON
1825 006752 000141 .WORD 141 ;ALL ERRORS TO TRAP TO EMT VECTOR
1826 006754 001213 .WORD MMUERR ;UNIQUE ERROR NUMBER
1827 ;ADDRESS OF ERROR MESSAGE
1828 006756 005037 177572 400$: CLR #177572 ;NO GO TO ERROR
;TURN MMU OFF
    
```

```

1829 006762 023727 000244 113672      CMP      @#244,#113672      ;IS TEST LOCATION CORRECT
1830 006770 001403                      BEQ      401$              ;YES GO ON
1831 006772 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1832 006774 000142                      .WORD   142               ;UNIQUE ERROR NUMBER
1833 006776 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1834                                     ;NO GO TO ERROR
1835 007000 020127 000244      401$:  CMP      R1,#244          ;IS R1 CORRECT
1836 007004 001403                      BEQ      402$              ;YES GO ON
1837 007006 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1838 007010 000143                      .WORD   143               ;UNIQUE ERROR NUMBER
1839 007012 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1840                                     ;NO GO TO ERROR
1841 007014 005037 177776      402$:  CLR      @#177776        ;SET PSW TO KERNEL MODE
1842 007020 022627 113672      CMP      (SP)+,#113672    ;IS STACK CORRECT
1843 007024 001403                      BEQ      403$              ;YES GO ON
1844 007026 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1845 007030 000144                      .WORD   144               ;UNIQUE ERROR NUMBER
1846 007032 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1847                                     ;NO GO TO ERROR
1848 007034 021627 177777      403$:  CMP      (SP),#177777 ;IS STACK CORRECT
1849 007040 001403                      BEQ      404$              ;YES GO ON
1850 007042 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1851 007044 000145                      .WORD   145               ;UNIQUE ERROR NUMBER
1852 007046 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1853                                     ;NO GO TO ERROR
1854 007050 005003                      CLR      R3                ;SETUP SOURCE FOR NEXT TEST
1855 007052 005237 177572      INC      @#177572        ;TURN MMU ON
1856 007056 106503                      MFPD     R3                ; TEST INSTRUCTION
1857 007060 022737 000004 177776      CMP      @4,@#177776     ;IS PSW CORRECT
1858 007066 001403                      BEQ      6$                ;YES GO ON
1859 007070 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1860 007072 000146                      .WORD   146               ;UNIQUE ERROR NUMBER
1861 007074 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1862                                     ;NO GO TO ERROR
1863 007076 005037 177572      6$:    CLR      @#177572        ;TURN MMU OFF
1864 007102 020327 000000      CMP      R3,#0           ;IS R3 CORRECT
1865 007106 001403                      BEQ      7$                ;YES GO ON
1866 007110 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1867 007112 000147                      .WORD   147               ;UNIQUE ERROR NUMBER
1868 007114 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1869                                     ;NO GO TO ERROR
1870 007116 022627 000000      7$:    CMP      (SP)+,#0      ;IS STACK CORRECT
1871 007122 001403                      BEQ      8$                ;YES GO ON
1872 007124 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1873 007126 000150                      .WORD   150               ;UNIQUE ERROR NUMBER
1874 007130 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1875                                     ;NO GO TO ERROR
1876 007132 022627 177777      8$:    CMP      (SP)+,#177777 ;IS STACK CORRECT
1877 007136 001403                      BEQ      9$                ;YES GO ON
1878 007140 104000                      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
1879 007142 000151                      .WORD   151               ;UNIQUE ERROR NUMBER
1880 007144 001213                      .WORD   MMUERR            ;ADDRESS OF ERROR MESSAGE
1881                                     ;NO GO TO ERROR
1882 007146 012637 000244      9$:    MOV      (SP)+,@#244    ;RESTORE TEST LOCATION
1883                                     ;
1884

```

```

1885 007152 TSM6C:
1886
1887
1888
1889 007152
1890 007152 005267 171626
1891 007156 005037 177572
1892 007162 005037 001042
1893 007166 012737 140000 177776
1894 007174 010637 001070
1895 007200 012737 040000 177776
1896 007206 010637 001066
1897 007212 012737 030000 177776
1898 007220 004767 172062
1899 007224 012737 000227 172516
1900 007232 013746 000244
1901 007236 012746 177777
1902 007242 012746 120413
1903 007246 012737 177777 000244
1904 007254 012767 077400 170336
1905 007262 012703 060244
1906 007266 005257 177572
1907 007272 006623
1908 007274 022737 030010 177776
1909 007302 001403
1910 007304 104000
1911 007306 000152
1912 007310 001213
1913
1914 007312 005037 177572 18:
1915 007316 012737 140000 177776
1916 007324 020637 001070
1917 007330 001403
1918 007332 104000
1919 007334 000153
1920 007336 001213
1921
1922 007340 012737 040000 177776 100:
1923 007346 020637 001066
1924 007352 001403
1925 007354 104000
1926 007356 000154
1927 007360 001213
1928
1929 007362 023727 000244 120413 200:
1930 007370 001403
1931 007372 104000
1932 007374 000155
1933 007376 001213
1934
1935 007400 020327 000246 28:
1936 007404 001403
1937 007406 104000
1938 007410 000156
1939 007412 001213
1940

```

```

*****
;TEST 11 TEST MPI (MOVE TO PREVIOUS INSTRUCTION SPACE)
*****
TST11:
INC 1TESTH ;INCREMENT TEST NUMBER
CLR 00177572 ;MMU OFF
CLR 00FLAG ;CLEAR MMU ABORT FLAG
MOV 0140000,00177776 ;POINT TO USER SPACE
MOV R6,00SAVUSE ;SAVE USER SP
MOV 040000,00177776 ;POINT TO SUPERVISOR SPACE
MOV R6,00SAVSUP ;SAVE SUPERVISOR SP
MOV 030000,00177776 ;SETUP PSW
JSR PC,MMU ;INIT MMU
MOV 027,00172516 ;SETUP MMR3
MOV 00244,-(SP) ;SAVE DATA AT TEST LOCATION
MOV 0177777,-(SP) ;PUT KNOWN DATA ON STACK
MOV 0120413,-(SP) ;PUT TEST DATA ON STACK
MOV 0177777,00244 ;PUT KNOWN DATA AT TEST LOCATION
MOV 077400,UPDRO ;SETUP UPDRO TO ABORT
MOV 0244,R3 ;SETUP POINTER TO TEST LOCATION
INC 00177572 ;TURN MMU ON
MPI (R3); ;TEST INSTRUCTION
CMP 030010,00177776 ;IS PSW CORRECT
BEQ 18 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 152 ;UNIQUE ERROR NUMBER
.WORD MMUERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
CLR 00177572 ;TURN MMU OFF
MOV 0140000,00177776 ;POINT TO USER SPACE
CMP R6,00SAVUSE ;IS USER SP CORRECT
BEQ 1001 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 153 ;UNIQUE ERROR NUMBER
.WORD MMUERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
MOV 040000,00177776 ;POINT TO SUPERVISOR SPACE
CMP R6,00SAVSUP ;IS SUPERVISOR SP CORRECT
BEQ 2001 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 154 ;UNIQUE ERROR NUMBER
.WORD MMUERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
CMP 00244,0120413 ;IS TEST LOCATION CORRECT
BEQ 28 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 155 ;UNIQUE ERROR NUMBER
.WORD MMUERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
CMP R3,0246 ;IS R3 CORRECT
BEQ 38 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 156 ;UNIQUE ERROR NUMBER
.WORD MMUERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR

```


1941	007414	005037	177776		3\$:	CLR	00177776		;SET PSW TO KERNEL MODE
1942	007420	021627	177777			CMP	(SP),0177777		;IS KERNEL STACK CORRECT
1943	007424	001403				BEQ	4\$;YES GO ON
1944	007426	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1945	007430	000157				.WORD	157		;UNIQUE ERROR NUMBER
1946	007432	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1947									;NO GO TO ERROR
1948	007434	012737	030017	177776	4\$:	MOV	030017,00177776		;SETUP PSW
1949	007442	012746	145121			MOV	0145121,-(SP)		;SETUP TEST DATA
1950	007446	012701	000244			MOV	0244,R1		;SETUP R1
1951	007452	005237	177572			INC	00177572		;TURN MMU ON
1952	007456	006611				MTPI	(R1)		;TEST INSTRUCTION
1953	007460	022737	030011	177776		CMP	030011,00177776		;IS PSW CORRECT
1954	007466	001403				BEQ	300\$;YES GO ON
1955	007470	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1956	007472	000160				.WORD	160		;UNIQUE ERROR NUMBER
1957	007474	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1958									;NO GO TO ERROR
1959	007476	005037	177572		300\$:	CLR	00177572		;TURN MMU OFF
1960	007502	023727	000244	145121		CMP	00244,0145121		;IS TEST LOCATION CORRECT
1961	007510	001403				BEQ	301\$;YES GO ON
1962	007512	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1963	007514	000161				.WORD	161		;UNIQUE ERROR NUMBER
1964	007516	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1965									;NO GO TO ERROR
1966	007520	020127	000244		301\$:	CMP	R1,0244		;IS R1 CORRECT
1967	007524	001403				BEQ	302\$;YES GO ON
1968	007526	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1969	007530	000162				.WORD	162		;UNIQUE ERROR NUMBER
1970	007532	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1971									;NO GO TO ERROR
1972	007534	005037	177776		302\$:	CLR	00177776		;SET PSW TO KERNEL MODE
1973	007540	021627	177777			CMP	(SP),0177777		;IS STACK CORRECT
1974	007544	001403				BEQ	304\$;YES GO ON
1975	007546	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1976	007550	000163				.WORD	163		;UNIQUE ERROR NUMBER
1977	007552	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1978									;NO GO TO ERROR
1979	007554	012737	030017	177776	304\$:	MOV	030017,00177776		;SETUP PSW
1980	007562	012746	122347			MOV	0122347,(SP)		;SETUP TEST DATA
1981	007566	012701	000246			MOV	0246,R1		;SETUP R1
1982	007572	005237	177572			INC	00177572		;TURN MMU ON
1983	007576	006641				MTPI	(R1)		;TEST INSTRUCTION
1984	007600	022737	030011	177776		CMP	030011,00177776		;IS PSW CORRECT
1985	007606	001403				BEQ	400\$;YES GO ON
1986	007610	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1987	007612	000164				.WORD	164		;UNIQUE ERROR NUMBER
1988	007614	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1989									;NO GO TO ERROR
1990	007616	005037	177572		400\$:	CLR	00177572		;TURN MMU OFF
1991	007622	023727	000244	122347		CMP	00244,122347		;IS TEST LOCATION CORRECT
1992	007630	001403				BEQ	401\$;YES GO ON
1993	007632	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1994	007634	000165				.WORD	165		;UNIQUE ERROR NUMBER
1995	007636	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
1996									;NO GO TO ERROR

TEST MTPI (MOVE TO PREVIOUS INSTRUCTION SPACE)

```

1997 007640 020127 000244      401$: CMP      R1,0244      ;IS R1 CORRECT
1998 007644 001403              BEQ      402$      ;YES GO ON
1999 007646 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2000 007650 000166              .WORD   166      ;UNIQUE ERROR NUMBER
2001 007652 001213              .WORD   MMUERR   ;ADDRESS OF ERROR MESSAGE
2002                                ;NO GO TO ERROR
2003 007654 005037 177776      402$: CLR      00177776 ;SET PSW TO KERNEL MODE
2004 007660 021627 177777      CMP      (SP),0177777 ;IS STACK CORRECT
2005 007664 001403              BEQ      404$      ;YES GO ON
2006 007666 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2007 007670 000167              .WORD   167      ;UNIQUE ERROR NUMBER
2008 007672 001213              .WORD   MMUERR   ;ADDRESS OF ERROR MESSAGE
2009                                ;NO GO TO ERROR
2010 007674 005046              CLR      -(SP)     ;SETUP STACK FOR NEXT TEST
2011 007676 005237 177572      INC      00177572 ;TURN MMU ON
2012 007702 006603              MTPI    R3        ;TEST INSTRUCTION
2013 007704 022737 000004 177776  CMP      04,00177776 ;IS PSW CORRECT
2014 007712 001403              BEQ      5$        ;YES GO ON
2015 007714 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2016 007716 000170              .WORD   170      ;UNIQUE ERROR NUMBER
2017 007720 001213              .WORD   MMUERR   ;ADDRESS OF ERROR MESSAGE
2018                                ;NO GO TO ERROR
2019 007722 005037 177572      5$:  CLR      00177572 ;TURN MMU OFF
2020 007726 020327 000000      CMP      R3,00     ;IS R3 CORRECT
2021 007732 001403              BEQ      6$        ;YES GO ON
2022 007734 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2023 007736 000171              .WORD   171      ;UNIQUE ERROR NUMBER
2024 007740 001213              .WORD   MMUERR   ;ADDRESS OF ERROR MESSAGE
2025                                ;NO GO TO ERROR
2026 007742 021627 177777      6$:  CMP      (SP)+,0177777 ;IS STACK CORRECT
2027 007746 001403              BEQ      7$        ;YES GO ON
2028 007750 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2029 007752 000172              .WORD   172      ;UNIQUE ERROR NUMBER
2030 007754 001213              .WORD   MMUERR   ;ADDRESS OF ERROR MESSAGE
2031                                ;NO GO TO ERROR
2032 007756 012637 000244      7$:  MOV      (SP)+,00244 ;RESTORE TEST LOCATION
2033
2034                                ;
2035 007762                                ;TSM6D:
2036                                ;*****
2037                                ;*TEST 12 TEST MTPD (MOVE TO PREVIOUS DATA SPACE)
2038                                ;*****
2039                                ;TST12:
2040 007762 005237 177016      INC      1TESTN    ;INCREMENT TEST NUMBER
2041 007766 005037 177572      CLR      00177572 ;MMU OFF
2042 007772 005037 001042      CLR      00FLAG   ;CLEAR MMU ABORT FLAG
2043 007776 012737 100000 177776  MOV      0140000,00177776 ;POINT TO USER SPACE
2044 010004 010637 001070      MOV      R6,00SAVUSE ;SAVE USER SP
2045 010010 012737 000000 177776  MOV      040000,00177776 ;POINT TO SUPERVISOR SPACE
2046 010016 010637 001066      MOV      R6,00SAVSUP ;SAVE SUPERVISOR SP
2047 010022 012737 030000 177776  MOV      030000,00177776 ;SETUP PSW
2048 010030 004707 171252      JSR     PC,MMU    ;INIT MMU
2049 010034 012737 000027 172516  MOV      027,00172516 ;SETUP MMR3
2050 010042 013745 000244      MOV      00244,-(SP) ;SAVE DATA AT TEST LOCATION
2051 010046 012737 177777      MOV      0177777,-(SP) ;PUT KNOWN DATA ON STACK
2052 010052 012746 100004      MOV      0100004,-(SP) ;PUT TEST DATA ON STACK

```

```

2053 010056 012737 177777 000244      MOV      0177777,00244      ;PUT KNOWN DATA AT TEST LOCATION
2054 010064 012707 077400 167506      MOV      077400,UIPDRO     ;SETUP UIPDRO TO ABORT
2055 010072 012703 000244              MOV      0244,R3          ;SETUP POINTER TO TEST LOCATION
2056 010076 005237 177572              INC      00177572         ;TURN MMU ON
2057 010102 106623 MTPD      (R3),           ;TEST INSTRUCTION
2058 010104 022737 030010 177776      CMP      030010,00177776  ;IS PSW CORRECT
2059 010112 001403      BEQ      1$              ;YES GO ON
2060 010114 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2061 010116 000173      .WORD   173             ;UNIQUE ERROR NUMBER
2062 010120 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2063                                     ;NO GO TO ERROR
2064 010122 005037 177572      CLR      00177572         ;TURN MMU OFF
2065 010126 012737 140000 177776 1$:    MOV      0140000,00177776  ;POINT TO USER SPACE
2066 010134 020637 001070      CMP      R6,00SAVUSE      ;IS USER SP CORRECT
2067 010140 001403      BEQ      100$           ;YES GO ON
2068 010142 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2069 010144 000173      .WORD   174             ;UNIQUE ERROR NUMBER
2070 010146 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2071                                     ;NO GO TO ERROR
2072 010150 012737 040000 177776 100$:  MOV      040000,00177776  ;POINT TO SUPERVISOR SPACE
2073 010156 020637 001066      CMP      R6,00SAVSUP      ;IS SUPERVISOR SP CORRECT
2074 010162 001403      BEQ      200$           ;YES GO ON
2075 010164 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2076 010166 000175      .WORD   175             ;UNIQUE ERROR NUMBER
2077 010170 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2078                                     ;NO GO TO ERROR
2079 010172 023727 000244 100004 200$:  CMP      00244,0100004    ;IS TEST LOCATION CORRECT
2080 010200 001403      BEQ      2$              ;YES GO ON
2081 010202 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2082 010204 000176      .WORD   176             ;UNIQUE ERROR NUMBER
2083 010206 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2084                                     ;NO GO TO ERROR
2085 010210 020327 000246      2$:    CMP      R3,0246          ;IS R3 CORRECT
2086 010214 001403      BEQ      3$              ;YES GO ON
2087 010216 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2088 010220 000177      .WORD   177             ;UNIQUE ERROR NUMBER
2089 010222 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2090                                     ;NO GO TO ERROR
2091 010224 005037 177776      3$:    CLR      00177776         ;SET PSW TO KERNEL MODE
2092 010230 021627 177777      CMP      (SP),0177777     ;IS KERNEL STACK CORRECT
2093 010234 001403      BEQ      4$              ;YES GO ON
2094 010236 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2095 010240 000200      .WORD   200             ;UNIQUE ERROR NUMBER
2096 010242 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2097                                     ;NO GO TO ERROR
2098 010244 012737 030017 177776 4$:    MOV      030017,00177776  ;SETUP PSW
2099 010252 012746 100737      MOV      0100737,-(SP)    ;SETUP TEST DATA
2100 010256 012701 000244      MOV      0244,R1          ;SETUP R1
2101 010262 005237 177572      INC      00177572         ;TURN MMU ON
2102 010266 106611 MTPD      (R1),           ;TEST INSTRUCTION
2103 010270 022737 030011 177776      CMP      030011,00177776  ;IS PSW CORRECT
2104 010276 001403      BEQ      300$           ;YES GO ON
2105 010300 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
2106 010302 000201      .WORD   201             ;UNIQUE ERROR NUMBER
2107 010304 001213      .WORD   MMUERR          ;ADDRESS OF ERROR MESSAGE
2108                                     ;NO GO TO ERROR

```

2109	010306	005037	177572		300\$:	CLR	00177572		;TURN MMU OFF
2110	010312	023727	000244	100737		CMP	00244,0100737		;IS TEST LOCATION CORRECT
2111	010320	001403				BEQ	301\$;YES GO ON
2112	010322	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2113	010324	000202				.WORD	202		;UNIQUE ERROR NUMBER
2114	010326	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2115									;NO GO TO ERROR
2116	010330	020127	000244		301\$:	CMP	R1,0244		;IS R1 CORRECT
2117	010334	001403				BEQ	302\$;YES GO ON
2118	010336	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2119	010340	000203				.WORD	203		;UNIQUE ERROR NUMBER
2120	010342	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2121									;NO GO TO ERROR
2122	010344	005037	177776		302\$:	CLR	00177776		;SET PSW TO KERNEL MODE
2123	010350	021627	177777			CMP	(SP),0177777		;IS STACK CORRECT
2124	010354	001403				BEQ	304\$;YES GO ON
2125	010356	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2126	010360	000204				.WORD	204		;UNIQUE ERROR NUMBER
2127	010362	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2128									;NO GO TO ERROR
2129	010364	012737	030017	177776	304\$:	MOV	030017,00177776		;SETUP PSW
2130	010372	012746	156711			MOV	0156711,-(SP)		;SETUP TEST DATA
2131	010376	012701	000246			MOV	0246,R1		;SETUP R1
2132	010402	005237	177572			INC	00177572		;TURN MMU ON
2133	010406	106631				MTPD	-(R1)		;TEST INSTRUCTION
2134	010410	020127	030011	177776		CMP	030011,00177776		;IS PSW CORRECT
2135	010416	001403				BEQ	400\$;YES GO ON
2136	010420	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2137	010422	000205				.WORD	205		;UNIQUE ERROR NUMBER
2138	010424	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2139									;NO GO TO ERROR
2140	010426	005037	177572		400\$:	CLR	00177572		;TURN MMU OFF
2141	010432	023727	000244	156711		CMP	00244,0156711		;IS TEST LOCATION CORRECT
2142	010440	001403				BEQ	401\$;YES GO ON
2143	010442	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2144	010444	000206				.WORD	206		;UNIQUE ERROR NUMBER
2145	010446	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2146									;NO GO TO ERROR
2147	010450	020127	000244		401\$:	CMP	R1,0244		;IS R1 CORRECT
2148	010454	001403				BEQ	402\$;YES GO ON
2149	010456	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2150	010460	000207				.WORD	207		;UNIQUE ERROR NUMBER
2151	010462	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2152									;NO GO TO ERROR
2153	010464	005037	177776		402\$:	CLR	00177776		;SET PSW TO KERNEL MODE
2154	010470	021627	177777			CMP	(SP),0177777		;IS STACK CORRECT
2155	010474	001403				BEQ	404\$;YES GO ON
2156	010476	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2157	010500	000210				.WORD	210		;UNIQUE ERROR NUMBER
2158	010502	001213				.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2159									;NO GO TO ERROR
2160	010504	005037	177572		404\$:	CLR	-(SP)		;SETUP STACK FOR NEXT TEST
2161	010506	005237				INC	00177572		;TURN MMU ON
2162	010512	106631				MTPD	R3		;TEST INSTRUCTION
2163	010514	021627	000004	177776		CMP	04,00177776		;IS PSW CORRECT
2164	010522	001403				BEQ	5\$;YES GO ON

```

2165 010524 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
2166 010526 000211          .WORD      211          ; UNIQUE ERROR NUMBER
2167 010530 001213          .WORD      MMUERR        ; ADDRESS OF ERROR MESSAGE
2168                                     ; NO GO TO ERROR
2169 010532 005037 177572    5$: CLR      @0177572      ; TURN MMU OFF
2170 010536 020327 000000    CMP      R3,#0          ; IS R3 CORRECT
2171 010542 001403          BEQ      6$             ; YES GO ON
2172 010544 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
2173 010546 000212          .WORD      212          ; UNIQUE ERROR NUMBER
2174 010550 001213          .WORD      MMUERR        ; ADDRESS OF ERROR MESSAGE
2175                                     ; NO GO TO ERROR
2176 010552 022627 177777    6$: CMP      (SP)+,#177777 ; IS STACK CORRECT
2177 010556 001403          BEQ      7$             ; YES GO ON
2178 010560 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
2179 010562 000213          .WORD      213          ; UNIQUE ERROR NUMBER
2180 010564 001213          .WORD      MMUERR        ; ADDRESS OF ERROR MESSAGE
2181                                     ; NO GO TO ERROR
2182 010566 012637 000244    7$: MOV      (SP)+,@0244  ; RESTORE TEST LOCATION
2183
2184                                     ;
2185 010572          TSMMU7:
2186                                     ; *****
2187                                     ; *TEST 13      TEST NON-RESIDENT ABORT
2188                                     ; *****
2189                                     ; TST13:
2190 010572 005267 170206    INC      $TESTN          ; INCREMENT TEST NUMBER
2191 010576 005037 177572    CLR      @0177572      ; MMU OFF
2192 010602 005067 170234    CLR      FLAG           ; CLEAR MMU ABORT FLAG
2193 010606 013746 000214    MOV      @0214,-(SP)    ; SAVE DATA AT TEST LOCATIONS
2194 010612 013746 000716    MOV      @0216,-(SP)
2195 010616 005067 170250    CLR      SAVMRO         ; CLEAR STATUS REGS SAVE AREAS
2196 010622 005067 170246    CLR      SAVMR1
2197 010626 005067 170244    CLR      SAVMR2
2198 010632 004767 170450    JSR      PC,MMU
2199 010636 012737 000000 177776 MOV      @30000,@0177776 ; SETUP PSW
2200 010644 012702 000200    MOV      @200,R2
2201 010650 012737 000700 177600 MOV      @77400,@0177600 ; SETUP FOR AN ABORT
2202 010656 004767 000164    JSR      PC,TS7
2203                                     ; CAUSE AN ABORT TO OCCUR AND
2204                                     ; THEN CHECK IF ABORT FLAG REGISTERED
2205                                     ; THIS EVENT AND CHECK IF STATUS REGS
2206                                     ; CONTAINED EXPECTED VALUES.
2207                                     ; IF NO ABORT OCCURRED THEN GO TO ERROR
2208 010662 012737 000700 177600 MOV      @77404,@0177600 ; OTHERWISE CONTINUE.
2209 010670 004767 000164    JSR      PC,TS7
2210                                     ; SETUP FOR AN ABORT
2211                                     ; CAUSE AN ABORT TO OCCUR AND
2212                                     ; THEN CHECK IF ABORT FLAG REGISTERED
2213                                     ; THIS EVENT AND CHECK IF STATUS REGS
2214                                     ; CONTAINED EXPECTED VALUES.
2215                                     ; IF NO ABORT OCCURRED THEN GO TO ERROR
2216 010674 012701 000220    MOV      @220,R1
2217 010700 004767 170402    JSR      PC,MMU
2218 010704 005003          CLR      R3
2219 010706 012767 000001 170126 MOV      @1,FLAG
2220 010714 012737 000001 177572 MOV      @1,@0177572
2221 010722 012737 100000 177776 MOV      @100000,@0177776 ; SETUP PSW FOR AN ABORT (ILLEGAL MODE)

```



```

2277 011156 001213          .WORD  MMUERR          ;ADDRESS OF ERROR MESSAGE
2278                                ;NOT OK THEN GO TO ERROR
2279 011160 005067 167706  OKAY7A: CLR  SAVMR0          ;CLEAR STATUS REGS SAVE AREAS
2280 011164 005067 167704          CLR  SAVMR1          ;
2281 011170 005067 167702          CLR  SAVMR2          ;
2282 011174 000207          RTS    PC              ;RETURN
2283
2284          ;ROUTINE TO CHECK IF A NONRESIDENT ABORT OCCURRED
2285
2286 011176 022767 0J0000 167636  TSM7:  CMP    #0,FLAG          ;DID AN ABORT OCCUR
2287 011204 001403          BEQ    TSMA          ;IF YES GO ON
2288 011206 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2289 011210 000220          .WORD  220          ;UNIQUE ERROR NUMBER
2290 011212 001213          .WORD  MMUERR        ;ADDRESS OF ERROR MESSAGE
2291
2292 011214 042737 040377 001072  TSMA:  BIC    #040377,#0SAVMR0 ;IF NO THEN GO TO ERROR
2293 011222 022767 100000 167642          CMP    #100000,SAVMR0 ;SETUP EXPECTED DATA
2294 011230 001403          BEQ    TSMB          ;TEST MMRO FOR EXPECTED VALUE
2295 011232 104000          ERROR          ;IF OK THEN CONTINUE
2296 011234 000221          .WORD  221          ;ALL ERRORS TO TRAP TO EMT VECTOR
2297 011236 001213          .WORD  MMUERR        ;UNIQUE ERROR NUMBER
2298
2299 011240 020367 167630  TSMB:  CMP    R3,SAVMR1        ;ADDRESS OF ERROR MESSAGE
2300 011244 001403          BEQ    TSMC          ;IF NO THEN GO TO ERROR
2301 011246 104000          ERROR          ;TEST MMR1 FOR EXPECTED VALUE
2302 011250 000222          .WORD  222          ;IF OK THEN CONTINUE
2303 011252 001213          .WORD  MMUERR        ;ALL ERRORS TO TRAP TO EMT VECTOR
2304
2305 011254 000207  TSMC:  RTS    PC              ;UNIQUE ERROR NUMBER
2306
2307 011256          ;ADDRESS OF ERROR MESSAGE
2308 011256          ;IF NOT OK THEN GO TO ERROR
2309
2310          ;RETURN
2311
2312          ;*****
2313          ;*TEST 14 TEST READ ONLY ABORTS
2314          ;*****
2315          TST14:
2316          INC    $TESTN          ;INCREMENT TEST NUMBER
2317          CLR    #0177572        ;MMU OFF
2318          CLR    FLAG          ;CLEAR MMU ABORT FLAG
2319          MOV    #0244,-(SP)      ;SAVE DATA AT TEST LOCATIONS
2320          MOV    #0246,-(SP)      ;
2321          CLR    SAVMR0          ;CLEAR STATUS REGS SAVE AREAS
2322          CLR    SAVMR1          ;
2323          CLR    SAVMR2          ;
2324          JSR    PC,MMU          ;INIT MMU
2325          MOV    #30000,#0177776 ;SETUP PSW
2326          MOV    #244,R2          ;
2327          MOV    #77402,#0177600 ;SETUP FOR AN ABORT
2328          MOV    #246,-(SP)      ;PUSH DATA ONTO THE STACK
2329          MOV    #1,FLAG          ;SETUP FLAG FOR AN ABORT
2330          MOV    #1,#0177572     ;TURN MMU ON
2331          MOV    R7,R1          ;SAVE PC
2332          MTPR  (R2),           ;CAUSE ABORT
2333          CMP    #0,FLAG          ;DID ABORT OCCUR
2334          BEQ    1$              ;IF YES THEN GO ON
2335
2336          ;IF NO THEN GO TO ERROR

```



```

2389
2390
2391
2392 011612 012703 012264      MOV    *PLF1,R3
2393 011616 012701 012334      MOV    *BN1,R1
2394 011622 012702 012402      MOV    *ABORT1,R2
2395 011626 004767 000030      JSR    PC,TSM9
2396
2397
2398
2399
2400
2401
2402 011632 005037 177572      CLR    *177572
2403 011636 012703 012274      MOV    *PLF1+10,R3
2404 011642 012701 012344      MOV    *BN1+10,R1
2405 011646 011337 177600      MOV    (R3),*177600
2406 011652 006521              MFPI   (R1)+
2407 011654 012605              MOV    (SP)+,R5
2408
2409 011656 000167 000566      JMP    TS9FIN
2410
2411
2412
2413 011662 012337 177600      ;ROUTINE TO CAUSE AND CHECK PAGE LENGTH ERROR ABORTS
2414 011666 010100              ;TSM9: MOV    (R3)+,*177600
2415 011670 012767 000001 167144  MOV    R1,R0
2416 011676 012737 000001 177572  MOV    *1,FLAG
2417 011704 010704              MOV    *1,*177572
2418 011706 006530              MOV    R7,R4
2419 011710 021227 000000      MFPI   *(R0)+
2420 011714 001011              CMP    (R2),*0
2421 011716 012605              BNC   2$
2422 011720 022767 000001 167114  MOV    (SP)+,R5
2423 011726 001403              CMP    *1,FLAG
2424 011730 104000              BEQ   1$
2425 011732 000230              ERROR
2426 011734 001213              .WORD 230
2427
2428 011736 000435              .WORD MMUERR
2429 011740 022767 000000 167074 1$: BR     6$
2430 011746 001403              2$: CMP    *0,FLAG
2431 011750 104000              BEQ   3$
2432 011752 000231              ERROR
2433 011754 001213              .WORD 231
2434
2435 011756 105067 167110              .WORD MMUERR
2436 011762 022767 040000 167102 3$: CLRB  SAVMRO
2437 011770 001403              CMP    *40000,SAVMRO
2438 011772 104000              BEQ   4$
2439 011774 000232              ERROR
2440 011776 001213              .WORD 232
2441
2442 012000 022767 000020 167066 4$: .WORD MMUERR
2443 012006 001403              CMP    *0,SAVMR1
2444 012010 104000              BEQ   5$
                ERROR

```

```

;CHECK TO SEE IF IT WAS SUPPOSED TO,
;AND IF YES CHECK ABORT FLAG AND
;STATUS REGISTERS.
;LET R3, R1, AND R2 POINT TO THE
;DOWNWARD EXPANSION TABLES
;
;TURN MMU ON
;DO RELOCATIONS FOR THE DIFFERENT
;VALUES OF THE PAGE LENGTH FIELD AND
;BLOCK NUMBER. IF AN ABORT OCCURS
;CHECK TO SEE IF IT WAS SUPPOSED TO,
;AND IF YES CHECK ABORT FLAG AND
;STATUS REGISTERS.
;MMU OFF
;POINT TO A VALUE WHICH SHOULD CAUSE
;AN ABORT IF MMU IS ON.
;SETUP UIPDRO
;DO A RELOCATION
;POP THE STACK
;
;SETUP UIPDRO
;SAVE A COPY OF R1
;SETUP FOR AN ABORT
;TURN MMU ON
;SAVE PC
;DO A RELOCATION OPERATION
;WAS AN ABORT SUPPOSED TO OCCUR
;IF YES GO TO 2$
;POP THE STACK
;DID AN ABORT OCCUR
;NO GO ON
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;YES GO TO ERROR
;
;DID AN ABORT OCCUR
;YES GO ON
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
;SETUP EXPECTED DATA
;TEST MMRO FOR EXPECTED VALUE
;IF OK THEN CONTINUE
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;NOT OK THEN GO TO ERROR
;TEST MMR1 FOR EXPECTED VALUE
;IF OK THEN CONTINUE
;ALL ERRORS TO TRAP TO EMT VECTOR

```

2445	012012	000233			.WORD	233		;UNIQUE ERROR NUMBER
2446	012014	001213			.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2447								;NOT OK THEN GO TO ERROR
2448	012016	020467	167054	5\$:	CMP	R4,SAVMR2		;TEST MMR2 FOR EXPECTED VALUE
2449	012022	001403			BEQ	6\$;IF OK THEN CONTINUE
2450	012024	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2451	012026	000234			.WORD	234		;UNIQUE ERROR NUMBER
2452	012030	001213			.WORD	MMUERR		;ADDRESS OF ERROR MESSAGE
2453								;NOT OK THEN GO TO ERROR
2454	012032	005067	167004	6\$:	CLR	FLAG		;CLEAR MMU ABORT FLAG
2455	012036	005067	167030		CLR	SAVMR0		;CLEAR STATUS REGS SAVE AREAS
2456	012042	005067	167026		CLR	SAVMR1		
2457	012046	005067	167024		CLR	SAVMR2		
2458	012052	005201			INC	R1		;POINT TO NEXT ENTRY
2459	012054	005201			INC	R1		
2460	012056	005202			INC	R2		
2461	012060	005202			INC	R2		
2462	012062	021327	000777		CMP	(R3),#777		;HAVE ALL ENTRIES BEEN TRIED
2463	012066	001275			BNE	TSM9		;NO REPEAT
2464	012070	000207			RTS	PC		;YES RETURN
2465								
2466								
2467								
2468	012072	070006						
2469	012074	070006						
2470	012076	070006						
2471	012100	013406						
2472	012102	020006						
2473	012104	004006						
2474	012106	040006						
2475	012110	070006						
2476	012112	024006						
2477	012114	004006						
2478	012116	014006						
2479	012120	012006						
2480	012122	002006						
2481	012124	001406						
2482	012126	004006						
2483	012130	002006						
2484	012132	000406						
2485	012134	007406						
2486	012136	001006						
2487	012140	003406						
2488	012142	000777						
2489	012144	013000						
2490	012146	016000						
2491	012150	017000						
2492	012152	002700						
2493	012154	014000						
2494	012156	002000						
2495	012160	004000						
2496	012162	007000						
2497	012164	002000						
2498	012166	000700						
2499	012170	004000						
2500	012172	001000						

;UPWARD EXPANSION TABLES

PLFO: .WORD 70006
. WORD 70006
. WORD 70006
. WORD 13406
. WORD 20006
. WORD 04006
. WORD 40006
. WORD 70006
. WORD 24006
. WORD 04006
. WORD 14006
. WORD 12006
. WORD 02006
. WORD 01406
. WORD 04006
. WORD 02006
. WORD 00406
. WORD 07406
. WORD 01006
. WORD 03406
. WORD 777
BNO: .WORD 013000
. WORD 016000
. WORD 017000
. WORD 002700
. WORD 014000
. WORD 002000
. WORD 004000
. WORD 007000
. WORD 002000
. WORD 000700
. WORD 004000
. WORD 001000

2501	012174	000300	.WORD	000300
2502	012176	000400	.WORD	000400
2503	012200	001400	.WORD	001400
2504	012202	000600	.WORD	000600
2505	012204	000200	.WORD	000200
2506	012206	001700	.WORD	001700
2507	012210	000300	.WORD	000300
2508	012212	000700	.WORD	000700
2509	012214	000000	ABORTO: .WORD	0
2510	012216	000000	.WORD	0
2511	012220	000001	.WORD	1
2512	012222	000000	.WORD	0
2513	012224	000001	.WORD	1
2514	012226	000001	.WORD	1
2515	012230	000000	.WORD	0
2516	012232	000000	.WORD	0
2517	012234	000000	.WORD	0
2518	012236	000000	.WORD	0
2519	012240	000001	.WORD	1
2520	012242	000000	.WORD	0
2521	012244	000000	.WORD	0
2522	012246	000001	.WORD	1
2523	012250	000001	.WORD	1
2524	012252	000001	.WORD	1
2525	012254	000001	.WORD	1
2526	012256	000000	.WORD	0
2527	012260	000001	.WORD	1
2528	012262	000000	.WORD	0
2529				
2530			;	DOWNWARD EXPANSION TABLES
2531			;	
2532	012264	000416	PLF1: .WORD	00416
2533	012266	020016	.WORD	20016
2534	012270	024016	.WORD	24016
2535	012272	034016	.WORD	34016
2536	012274	074016	.WORD	74016
2537	012276	040016	.WORD	40016
2538	012300	020016	.WORD	20016
2539	012302	000016	.WORD	00016
2540	012304	030016	.WORD	30016
2541	012306	010016	.WORD	10016
2542	012310	014016	.WORD	14016
2543	012312	004016	.WORD	04016
2544	012314	002016	.WORD	02016
2545	012316	000416	.WORD	00416
2546	012320	000016	.WORD	00016
2547	012322	003416	.WORD	03416
2548	012324	001016	.WORD	01016
2549	012326	001416	.WORD	01416
2550	012330	000416	.WORD	00416
2551	012332	000777	.WORD	777
2552	012334	000100	BN1: .WORD	000100
2553	012336	010000	.WORD	010000
2554	012340	006000	.WORD	006000
2555	012342	016000	.WORD	016000
2556	012344	016000	.WORD	016000

2557	012346	004000	.WORD	004000
2558	012350	000000	.WORD	000000
2559	012352	000000	.WORD	000000
2560	012354	004000	.WORD	004000
2561	012356	004000	.WORD	004000
2562	012360	004000	.WORD	004000
2563	012362	000000	.WORD	000000
2564	012364	000300	.WORD	000300
2565	012366	000000	.WORD	000000
2566	012370	000400	.WORD	000400
2567	012372	001000	.WORD	001000
2568	012374	000100	.WORD	000100
2569	012376	000400	.WORD	000400
2570	012400	000200	.WORD	000200
2571	012402	000000	.WORD	0
2572	012404	000000	.WORD	0
2573	012406	000000	.WORD	0
2574	012410	000000	.WORD	0
2575	012412	000001	.WORD	1
2576	012414	000001	.WORD	1
2577	012416	000001	.WORD	1
2578	012420	000000	.WORD	0
2579	012422	000001	.WORD	1
2580	012424	000000	.WORD	0
2581	012426	000000	.WORD	0
2582	012430	000001	.WORD	1
2583	012432	000001	.WORD	1
2584	012434	000001	.WORD	1
2585	012436	000000	.WORD	0
2586	012440	000000	.WORD	0
2587	012442	000001	.WORD	1
2588	012444	000000	.WORD	0
2589	012446	000000	.WORD	0

ABORT1:

2590				
2591	012450			
2592	012450			
2593				
2594				
2595				
2596	012450			
2597	012450	005267	166330	
2598	012454	005037	177572	
2599	012460	005067	166356	
2600	012464	005067	166402	
2601	012470	005067	166400	
2602	012474	005067	166376	
2603	012500	004767	166602	
2604	012504	005037	177776	
2605	012510	012702	020200	
2606	012514	012737	077400	172302
2607	012522	012767	000001	166312
2608	012530	012737	000001	177572
2609	012536	010701		
2610	012540	006522		
2611	012542	012704	100003	
2612	012546	004767	000210	

```

;
TS9FIN:
TSM10:
;*****
; *TEST 16      FUNCTIONAL TEST OF BITS <6:1> OF MMRO
;*****
TST16:
    INC     $TESTN          ; INCREMENT TEST NUMBER
    CLR     @#177572       ; MMU OFF
    CLR     FLAG           ; CLEAR MMU ABORT FLAG
    CLR     SAVMRO         ; CLEAR STATUS PEGS SAVE AREAS
    CLR     SAVMR1
    CLR     SAVMR2
    JSR     PC,MMU        ; INIT MMU
    CLR     @#177776       ; INIT PSW: PREVIOUS MODE = KERNAL
    MOV     @20200,R2
    MOV     @77400,@#172302 ; SETUP KIPDR1 TO ABORT
    MOV     @1,FLAG       ; SETUP FLAG FOR AN ABORT
    MOV     @1,@#177572   ; TURN MMU ON
    MOV     R7,R1         ; SAVE PC
    MFPI    (R2)+         ; DO A RELOCATION VIA KIPAR1
    MOV     @100003,R4    ; SETUP EXPECTED DATA
    JSR     PC,TS10       ; CHECK IF AN ABORT OCCURRED AND

```



```

2669
2670 013014 022767 000022 166052 2$: CMP    #22,SAVMR1      ;NO GO TO ERROR
2671 013022 001403          BEQ    3$           ; TEST MMR1 FOR EXPECTED DATA
2672 013024 104000          ERROR          ;OK GO ON
2673 013026 000237          .WORD  237       ;ALL ERRORS TO TRAP TO ENT VECTOR
2674 013030 001213          .WORD  MMUERR    ;UNIQUE ERROR NUMBER
2675                                     ;ADDRESS OF ERROR MESSAGE
2676 013032 020167 166040 3$: CMP    R1,SAVMR2      ;NO GO TO ERROR
2677 013036 001403          BEQ    4$           ; TEST MMR2 FOR EXPECTED DATA
2678 013040 104000          ERROR          ;OK GO ON
2679 013042 000240          .WORD  240       ;ALL ERRORS TO TRAP TO ENT VECTOR
2680 013044 001213          .WORD  MMUERR    ;UNIQUE ERROR NUMBER
2681                                     ;ADDRESS OF ERROR MESSAGE
2682 013046 005067 166020 4$: CLR    SAVMR0      ;NO GO TO ERROR
2683 013052 005067 166016          CLR    SAVMR1    ; CLEAR MMU STATUS REGS SAVE AREAS
2684 013056 005067 166014          CLR    SAVMR2    ;
2685 013062 000207          RTS     PC        ;RETURN
2686
2687 013064          ;
2688 013064          ;T10FIN:
2689          ;TSM11:
2690          ;*****
2691          ;*TEST 17      TEST DATA SPACE BITS MMR3
2692          ;*****
2693          ;TST17:
2694          ;
2695          ;INC    #TESTN          ;INCREMENT TEST NUMBER
2696          ;CLR    #0177572      ;MMU OFF
2697          ;CLR    FLAG          ;CLEAR MMU ABORT FLAG
2698          ;MOV    #30000,#0177776 ;SETUP PSW
2699          ;MOV    #26,R1        ;SETUP FIRST MMR3 VALUE
2700          ;MOV    #177610,R3   ;POINT TO UIPDR4
2701          ;MOV    #21,R4        ;SETUP SECOND MMR3 VALUE
2702          ;JSR    PC,TS11      ; TEST ENABLE USER DATA SPACE BIT
2703          ;MOV    #0,#0177776   ;SETUP PSW
2704          ;MOV    #23,R1        ;SETUP FIRST MMR3 VALUE
2705          ;MOV    #172310,R3   ;POINT TO KIPDR4
2706          ;JSR    PC,TS11      ;SETUP SECOND MMR3 VALUE
2707          ;MOV    #10000,#0177776 ; TEST ENABLE KERNEL DATA SPACE BIT
2708          ;MOV    #25,R1        ;SETUP PSW
2709          ;MOV    #172210,R3   ;SETUP FIRST MMR3 VALUE
2710          ;MOV    #22,R4        ;POINT TO SIPDR4
2711          ;JSR    PC,TS11      ;SETUP SECOND MMR3 VALUE
2712          ; TEST ENABLE SUPERVISOR DATA SPACE BIT
2713          ;
2714          ;JMP    T11FIN
2715          ;
2716          ;ROUTINE TO TEST ENABLE DATA SPACE BITS OF MMR3
2717          ;
2718          ;TS11: JSR    PC,MMU          ;INIT MMU
2719          ;MOV    R1,#0172516    ;DISABLE DATA SPACE OF MODE UNDER TEST
2720          ;MOV    #77400,(R3)   ;SETUP IPDR TO ABORT
2721          ;MOV    #100000,R2    ;
2722          ;MOV    #1,FLAG       ;SETUP FLAG FOR AN ABORT
2723          ;MOV    #1,#0177572   ;MMU ON
2724          ;MFPD (R2)           ;DO A RELOCATION
2725          ;CMP    #0,FLAG       ;DID AN ABORT OCCUR
2726          ;BEQ    1$           ;YES GO ON

```



```

2837
2838 013742
2839 013742 005267 165036
2840 013746 005037 177572
2841 013752 005067 165064
2842 013756 005037 177776
2843 013762 004767 165320
2844 013766 012737 000020 172516
2845 013774 012703 014164
2846 014000 012701 014216
2847 014004 012133 1$:
2848 014006 021127 000333
2849 014012 001374
2850 014014 012703 014302
2851 014020 012701 014250
2852 014024 012702 014334
2853 014030 012237 177776 2$:
2854 014034 013305
2855 014036 012737 000001 177572
2856 014044 006531
2857 014046 012604
2858 014050 005037 177572
2859 014054 020504
2860 014056 001403
2861 014060 104000
2862 014062 000246
2863 014064 001213
2864
2865 014066 021327 000111 3$:
2866 014072 001356
2867 014074 005203
2868 014076 005203
2869 014100 005201
2870 014102 005201
2871 014104 005202
2872 014106 005202
2873 014110 012237 177776
2874 014114 013305
2875 014116 012737 000027 172516
2876 014124 012737 000001 177572
2877 014132 106531
2878 014134 012604
2879 014136 005037 177572
2880 014142 005037 172516
2881 014146 020504
2882 014150 001403
2883 014152 104000
2884 014154 000247
2885 014156 001213
2886
2887 014160 4$:
2888
2889 014160 000167 000202
2890
2891
2892

```

```

*****
TST21:
INC $TESTN ; INCREMENT TEST NUMBER
CLR @0177572 ; MMU OFF
CLR FLAG ; CLEAR MMU ABORT FLAG
CLR @0177776 ; INIT PSW
JSR PC,MMU ; INIT MMU
MOV @20,@0172516 ; INIT MMR3
MOV @PARAD1,R3 ; SETUP PARS WITH TEST VALUES
MOV @PARVA1,R1
MOV (R1)+,@(R3)+
CMP (R1),@333
BNE 1$
MOV @PHY1,R3 ; SET POINTERS TO ADDER PART A
MOV @VIR1,R1 ; TEST TABLES.
MOV @MODE1,R2
MOV (R2)+,@0177776 ; INIT PSW
MOV @R3)+,R5 ; SAVE DATA AT PHYSICAL ADDRESS
MOV @1,@0177572 ; TURN MMU ON
MFPI @R1)+ ; SAVE DATA AT RELOCATED VIRTUAL ADDRESS
MOV (SP)+,R4
CLR @0177572
CMP R5,R4 ; TURN MMU OFF
BEQ 3$ ; IS DATA EQUAL TO EXPECTED
ERROR ; YES GO ON
.WORD 246 ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD MMUERR ; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; NO IT IS AN ADDER ERROR
CMP (R3),@111 ; ARE WE READY TO TEST DATA SPACE
BNE 2$ ; NO GO TO 2$
INC R3 ; POINT TO DATA SPACE VALUES
INC R3
INC R1
INC R1
INC R2
INC R2
MOV (R2)+,@0177776 ; INIT PSW
MOV @R3)+,R5 ; SAVE DATA AT PHYSICAL ADDRESS
MOV @27,@0172516 ; INIT MMR3
MOV @1,@0177572 ; TURN MMU ON
MFPI @R1)+ ; SAVE DATA AT RELOCATED VIRTUAL ADDRESS
MOV (SP)+,R4 ; POP THE STACK
CLR @0177572 ; TURN MMU OFF
CLR @0172516 ; CLEAR MMR3
CMP R5,R4 ; IS DATA EQUAL TO EXPECTED
BEQ 4$ ; YES GO ON
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 247 ; UNIQUE ERROR NUMBER
.WORD MMUERR ; ADDRESS OF ERROR MESSAGE
; NO IT IS AN ADDER ERROR
JMP T13FTN
;
; ADDER TEST PART A TABLES
;

```

2893	014164	172240	PARAD1:	.WORD	172240
2894	014166	177642		.WORD	177642
2895	014170	172252		.WORD	172252
2896	014172	177640		.WORD	177640
2897	014174	172242		.WORD	172242
2898	014176	172254		.WORD	172254
2899	014200	177652		.WORD	177652
2900	014202	177644		.WORD	177644
2901	014204	172246		.WORD	172246
2902	014206	177654		.WORD	177654
2903	014210	172250		.WORD	172250
2904	014212	177660		.WORD	177660
2905	014214	000333		.WORD	333
2906	014216	000000	PARVA1:	.WORD	000000
2907	014220	000010		.WORD	000010
2908	014222	177777		.WORD	177777
2909	014224	177601		.WORD	177601
2910	014226	000010		.WORD	000010
2911	014230	000052		.WORD	000052
2912	014232	000070		.WORD	000070
2913	014234	000010		.WORD	000010
2914	014236	000010		.WORD	000010
2915	014240	000060		.WORD	000060
2916	014242	000000		.WORD	000000
2917	014244	000010		.WORD	000010
2918	014246	000333		.WORD	333
2919	014250	000000	VIR1:	.WORD	000000
2920	014252	025000		.WORD	025000
2921	014254	135224		.WORD	135224
2922	014256	017700		.WORD	017700
2923	014260	033000		.WORD	033000
2924	014262	145252		.WORD	145252
2925	014264	121000		.WORD	121000
2926	014266	043000		.WORD	043000
2927	014270	075000		.WORD	075000
2928	014272	142000		.WORD	142000
2929	014274	117700		.WORD	117700
2930	014276	000111		.WORD	111
2931	014300	007000		.WORD	007000
2932	014302	000000	PHY1:	.WORD	000000
2933	014304	006000		.WORD	006000
2934	014306	015124		.WORD	015124
2935	014310	000000		.WORD	000000
2936	014312	014000		.WORD	014000
2937	014314	012452		.WORD	012452
2938	014316	010000		.WORD	010000
2939	014320	004000		.WORD	004000
2940	014322	016000		.WORD	016000
2941	014324	010000		.WORD	010000
2942	014326	017700		.WORD	017700
2943	014330	000111		.WORD	111
2944	014332	010000		.WORD	010000
2945	014334	010000	MODE 1:	.WORD	010000
2946	014336	030000		.WORD	030000
2947	014340	010000		.WORD	010000
2948	014342	030000		.WORD	030000

```

3117 015312 005037 177572      CLR      @#177572      ;TURN MMU OFF
3118 015316 104000      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
3119 015320 000256      .WORD   256          ;UNIQUE ERROR NUMBER
3120 015322 001213      .WORD   MMUERR      ;ADDRESS OF ERROR MESSAGE
3121                                ;NOT EQUAL GO TO ERROR
3122 015324 005037 177572      CLR      @#177572      ;TURN MMU OFF
3123 015330 021427 000333      CMP      (R4),#333    ;ARE WE DONE
3124 015334 001351      BNE     1$           ;NO GO TO 1$
3125 015336 012704 015704      MOV     @PARVA2,R4    ;SET POINTERS TO PAR INIT TABLES
3126 015342 012701 015652      MOV     @PARAD2,R1    ;
3127 015346 012431      3$:  MOV     (R4)+,@(R1)+ ;INIT PARS
3128 015350 021127 000333      CMP     (R1),#333    ;ARE WE DONE
3129 015354 001374      BNE     3$           ;NO, GO TO 3$
3130 015356 012704 015770      MOV     @MODE2,R4    ;SET POINTERS TO ADDER PART B TABLES
3131 015362 012701 015736      MOV     @VIR2,R1    ;
3132 015366 012702 016022      MOV     @PARVA3,R2    ;
3133 015372 012703 016054      MOV     @VIR3,R3    ;
3134 015376 004767 000076      4$:  JSR     PC,TS14    ;WRITE DATA TO PHYSICAL ADDRESS AND THEN
3135                                ;CHECK IF DATA AT PHYSICAL ADDRESS IS
3136                                ;EQUAL TO EXPECTED AND IF NOT DETERMINE
3137                                ;IF IT IS AN ADDER ERROR OR A MEMORY ERROR
3138 015402 021127 000111      CMP     (R1),#111    ;HAVE WE DONE ALL THE 22 BIT MODE I SPACE
3139                                ;CASES
3140                                ;NO GO TO 4$
3141 015406 001373      BNE     4$           ;POINT TO 22 BIT MODE D SPACE CASE
3142 015410 005201      INC     R1           ;
3143 015412 005201      INC     R1           ;
3144 015414 005204      INC     R4           ;
3145 015416 005204      INC     R4           ;
3146 015420 012737 000027 172516      MOV     @27,@#172516 ;INIT MMR3
3147 015426 012437 177776      MOV     (R4)+,@#177776 ;INIT PSW
3148 015432 012746 052525      MOV     @52525,-(SP) ;PUSH DATA ONTO STACK
3149 015436 012737 000001 177572      MOV     @1,@#177572  ;TURN MMU ON
3150 015444 106631      MTPD   @(R1)+       ;WRITE DATA TO PHYSICAL ADDRESS
3151 015446 005037 177572      CLR     @#177572    ;TURN MMU OFF
3152 015452 012737 000020 172516      MOV     @20,@#172516 ;INIT MMR3
3153 015460 004767 000040      JSR     PC,T14      ;CHECK IF DATA AT PHYSICAL ADDRESS IS EQUAL
3154                                ;TO EXPECTED AND IF NOT DETERMINE IF IT
3155                                ;IS AN ADDER ERROR OR A MEMORY ERROR
3156 015464 005037 172516      CLR     @#172516    ;INIT MMR3 FOR 18 BIT MODE
3157 015470 004767 000004      JSR     PC,TS14    ;WRITE DATA TO PHYSICAL ADDRESS AND THEN
3158                                ;CHECK IF DATA AT PHYSICAL ADDRESS IS
3159                                ;EQUAL TO EXPECTED AND IF NOT DETERMINE IF
3160                                ;IT IS AN ADDER ERROR OR A MEMORY ERROR
3161 015474 000167 000406      JMP     T14FIN
3162                                ;
3163                                ;ROUTINE TO WRITE DATA TO PHYSICAL ADDRESS AND TO CHECK IF DATA AT
3164                                ;PHYSICAL ADDRESS IS EQUAL TO EXPECTED AND IF NOT DETERMINE IF IT IS
3165                                ;AN ADDER ERROR OR A MEMORY ERROR
3166                                ;
3167 015500 012437 177776      TS14: MOV     (R4)+,@#177776 ;INIT PSW
3168 015504 012737 000001 177572      MOV     @1,@#177572  ;TURN MMU ON
3169 015512 012746 052525      MOV     @52525,-(SP) ;WRITE DATA ONTO STACK
3170 015516 006631      MTPD   @(R1)+       ;WRITE DATA TO PHYSICAL ADDRESS VIA STACK
3171 015520 005037 177572      CLR     @#177572    ;TURN MMU OFF
3172 015524 012737 010000 177776      T14:  MOV     @10000,@#177776 ;INIT PSW

```

```

3285 016104 000333          .WORD 333
3286
3287
3288
3289 016106          ;
3290          T14FIN:
3291          ;:*****
3292          ;*TEST 24      TEST NON-EXISTANT MEMORY TRAP
3293          ;:*****
3294          ;WE ARE ASSUMING THAT THE NON-EXISTANT MEMORY TIME OUT
3295          ;FEATURE IS WORKING SINCE WE CAN'T GUARANTEE THAT
3296          ;THE SYSTEM BEING TESTED HAS A NON-EXISTANT MEMORY LOCATION.
3297          ;AT THIS TIME WE WILL ATTEMPT TO TEST THE NXM FUNCTION
3298          ;:*****
3299 016106 005267 162672          TST24:
3300 016112 004767 163170          INC      $TESTN          ; INCREMENT TEST NUMBER
3301 016116 012737 177400 172354  JSR      PC,MMU          ; INIT THE MMU
3302 016124 016767 161654 162676  MOV     @177400,@KIPAR6  ; SET KIPAR6 TO RELOCATE TO HIGHEST MEMORY
3303 016132 016767 000026 161644  MOV     4,SLOC00        ; SAVE VECTOR
3304 016140 052767 000001 161424  MOV     2$,4            ; LOAD VEC WITH ADDR OF TRAP HANDLER
3305 016146 005067 161614          BIS     @BIT00,SRO      ; TURN ON THE MMU
3306 016152 005067 161620          CLR     CPREG           ; CLEAR THE CPU ERROR REGISTER
3307 016156 005737 157776          CLR     PS             ; CLEAR THE PSW
3308 016162 000423          TST     @0157776       ; ACCESS PHYSICAL ADDR 17757776
3309          1$: BR      NXMFIN          ; IF IT DOESN'T TRAP WE'LL ASSUME
3310          ; THAT THIS IS A 4 MEGABYTE SYSTEM
3311 016164 022767 000040 161574 2$: CMP     @BIT05,CPREG    ; AND GO TO THE NEXT TEST
3312 016172 001403          BEQ     3$            ; IS CPU ERROR REGISTER CORRECT?
3313 016174 104000          ;
3314 016176 000261          ; ALL ERRORS TO TRAP TO EMT VECTOR
3315 016200 001213          .WORD 261            ; UNIQUE ERROR NUMBER
3316 016202 022726 016162 3$: CMP     @1$,(SP)+      ; ADDRESS OF ERROR MESSAGE
3317 016206 001403          BEQ     4$            ; IS CONTENTS OF STACK CORRECT?
3318 016210 104000          ;
3319 016212 000262          ; ALL ERRORS TO TRAP TO EMT VECTOR
3320 016214 001213          .WORD 262            ; UNIQUE ERROR NUMBER
3321 016216 022726 000000 4$: CMP     @0,(SP)+      ; ADDRESS OF ERROR MESSAGE
3322 016222 001403          BEQ     NXMFIN        ; IS CONTENTS OF STACK CORRECT?
3323 016224 104000          ;
3324 016226 000263          ; ALL ERRORS TO TRAP TO EMT VECTOR
3325 016230 001213          .WORD 263            ; UNIQUE ERROR NUMBER
3326 016232 005067 161334  NXMFIN: CLR     SRO      ; ADDRESS OF ERROR MESSAGE
3327 016236 005067 161524          CLR     CPREG         ; TURN OFF THE MMU
3328 016242 016767 162562 161534  MOV     SLOC00,4      ; CLEAR THE CPU ERROR REGISTER
3329          ; RESTORE THE VECTOR
3330
3331 016250          TSMM15:
3332          ;:*****
3333          ;*TEST 25      PAGE WRITTEN BIT TEST
3334          ;:*****
3335          TST25:
3336 016250 005267 162530          INC     $TESTN          ; INCREMENT TEST NUMBER
3337 016254 005037 177572          CLR     @0177576      ; MMU OFF
3338 016260 005067 162556          CLR     FLAG           ; CLEAR MMU ABORT FLAG
3339 016264 004767 163016          JSR     PC,MMU         ; INIT MMU
3340 016270 005037 177776          CLR     @0177776      ; INIT PSW

```

```

3341 016274 012704 172300      MOV    #172300,R4      ;SET POINTER TO KPDRS
3342 016300 004767 000114      JSR    PC,TS15        ;DO RELOCATIONS AND TEST KIPDRS FOR
3343                                     ;PAGE WRITTEN BIT BEING SET AND IF
3344                                     ;NOT SET GO TO ERROR
3345 016304 004767 000164      JSR    PC,T15        ;DO RELOCATIONS AND TEST KPDRS FOR
3346                                     ;PAGE WRITTEN BIT BEING SET AND IF NOT
3347                                     ;SET GO TO ERROR
3348 016310 012737 050000 177776  MOV    #50000,#0177776 ;INIT PSW
3349 016316 012704 172200      MOV    #172200,R4    ;SET POINTER TO SPDRS
3350 016322 004767 000072      JSR    PC,TS15        ;DO RELOCATIONS AND TEST SIPDRS FOR
3351                                     ;PAGE WRITTEN BIT BEING SET AND IF NOT
3352                                     ;SET GO TO ERROR
3353 016326 004767 000142      JSR    PC,T15        ;DO RELOCATIONS AND TEST SPDRS FOR
3354                                     ;PAGE WRITTEN BIT BEING SET AND IF NOT
3355                                     ;SET GO TO ERROR
3356 016332 005037 177776      CLR    #0177776      ;INIT PSW TO A KNOWN STATE
3357 016336 012737 170000 177776  MOV    #170000,#0177776 ;INIT PSW
3358 016344 012704 177600      MOV    #177600,R4    ;SET POINTER TO UPDRS
3359 016350 004767 000044      JSR    PC,TS15        ;DO RELOCATIONS AND TEST UIPDRS FOR
3360                                     ;PAGE WRITTEN BIT BEING SET AND IF
3361                                     ;NOT SET GO TO ERROR
3362 016354 004767 000114      JSR    PC,T15        ;DO RELOCATIONS AND TEST UPDRS FOR
3363                                     ;PAGE WRITTEN BIT BEING SET AND IF NOT
3364                                     ;SET GO TO ERROR
3365 016360 005037 177776      CLR    #0177776      ;INIT PSW TO A KNOWN STATE
3366 016364 012704 172300      MOV    #172300,R4    ;SET POINTER TO KPDRS
3367 016370 004767 000162      JSR    PC,T15A       ;EXPLICITLY WRITE TO KPDRS AND TEST
3368                                     ;FOR PAGE WRITTEN BIT BEING CLEARED
3369                                     ;AND IF NOT CLEARED GO TO ERROR
3370 016374 012704 172200      MOV    #172200,R4    ;SET POINTER TO SPDRS
3371 016400 004767 000152      JSR    PC,T15A       ;EXPLICITLY WRITE TO SPDRS AND TEST
3372                                     ;FOR PAGE WRITTEN BIT BEING CLEARED
3373                                     ;AND IF NOT CLEARED GO TO ERROR
3374 016404 012704 177600      MOV    #177600,R4    ;SET POINTER TO UPDRS
3375 016410 004767 000142      JSR    PC,T15A       ;EXPLICITLY WRITE TO UPDRS AND TEST
3376                                     ;FOR PAGE WRITTEN BIT BEING CLEARED
3377                                     ;AND IF NOT CLEARED GO TO ERROR
3378
3379 016414 000167 000170      JMP    T15FIN
3380
3381      ;ROUTINE TO DO RELOCATIONS AND TEST IPDRS FOR PAGE WRITTEN BIT BEING
3382      ;SET AND IF NOT SET REPORT AN ERROR
3383
3384 016420 005001      TS15: CLR    R1      ;SET POINTER TO VIRTUAL ADDRESS
3385 016422 012737 000020 172516  MOV    #20,#0172516  ;INIT MMR3
3386 016430 012737 000001 177572 14: MOV    #1,#0177572  ;TURN MMU ON
3387 016436 011111      MOV    (R1),(R1)    ;DO A RELOCATION
3388 016440 005037 177572      CLR    #0177572    ;TURN MMU OFF
3389 016444 022427 077506      CMP    (R4),#077506 ;IS DATA EQUAL TO EXPECTED
3390 016450 001403      BEQ    #1          ;OK GO ON
3391 016452 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
3392 016454 000264      .WORD 264        ;UNIQUE ERROR NUMBER
3393 016456 001213      .WORD MMUERR    ;ADDRESS OF ERROR MESSAGE
3394
3395 016460 062701 020000 23: ADD    #20000,R1    ;NO GO TO ERROR
3396 016464 020127 160000      CMP    R1,#160000  ;POINT TO NEXT VIRTUAL ADDRESS
                                     ;ARE WE DONE

```

```

3397 016470 001357          BNE 1$          ;NO GO TO 1$
3398 016472 000207          RTS   PC        ;RETURN
3399
3400          ;ROUTINE TO DO RELOCATIONS AND TEST DPDRS FOR PAGE WRITTEN BIT BEING SET
3401          ;AND IF NOT SET REPORT AN ERROR
3402
3403 016474 005001          T15:  CLR  R1          ;SET POINTER TO VIRTUAL ADDRESS
3404 016476 062704 000002    ADD  #2,R4        ;POINT TO FIRST DPDR
3405 016502 012737 000027 172516  MOV  #27,#0172516 ;INIT MMR3
3406 016510 012737 000001 177572 1$:  MOV  #1,#0177572 ;TURN MMU ON
3407 016516 011146          MOV  (R1),-(SP)  ;PUSH DATA ONTO THE STACK
3408 016520 106611          MTPD (R1)        ;DO A RELOCATION
3409 016522 005037 177572    CLR  #0177572   ;TURN MMU OFF
3410 016526 022427 077506    CMP  (R4),#077506 ;IS DATA EQUAL TO EXPECTED
3411 016532 001403          BEQ  2$          ;OK GO ON
3412 016534 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3413 016536 000265          .WORD 265       ;UNIQUE ERROR NUMBER
3414 016540 001213          .WORD MMUERR    ;ADDRESS OF ERROR MESSAGE
3415
3416 016542 062701 020000    2$:  ADD  #20000,R1  ;NO GO TO ERROR
3417 016546 020127 160000    CMP  R1,#160000 ;POINT TO NEXT VIRTUAL ADDRESS
3418 016552 001356          BNE  1$          ;ARE WE DONE
3419 016554 000207          RTS   PC        ;NO GO TO 1$
3420          ;RETURN
3421          ;ROUTINE TO EXPLICITLY WRITE TO PDRS AND TEST PAGE WRITTEN BIT FOR BEING
3422          ;CLEARED AND IF NOT CLEARED REPORT AN ERROR
3423
3424 016556 005002          T15A: CLR  R2          ;CLEAR COUNTER
3425 016560 011414          1$:  MOV  (R4),(R4)  ;DO AN EXPLICIT WRITE TO PDR
3426 016562 022427 077406    CMP  (R4),#077406 ;IS DATA EQUAL TO EXPECTED
3427 016566 001403          BEQ  2$          ;OK GO ON
3428 016570 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3429 016572 000266          .WORD 266       ;UNIQUE ERROR NUMBER
3430 016574 001213          .WORD MMUERR    ;ADDRESS OF ERROR MESSAGE
3431
3432 016576 005202          2$:  INC  R2          ;NO GO TO ERROR
3433 016600 020227 000020    CMP  R2,#20     ;INCREMENT POINTER
3434 016604 001365          BNE  1$          ;ARE WE DONE
3435 016606 000207          RTS   PC        ;NO GO TO 1$
3436 016610          ;RETURN
3437
3438 016610          T15FIN:
3439          ;TSM16:
3440          ;*****
3441          ;*TEST 26 TEST CSM (CALL SUPERVISOR MODE)
3442          ;*****
3442 016610          TST26:
3443 016610 005267 162170    INC  #TESTN     ;INCREMENT TEST NUMBER
3444 016614 005037 177572    CLR  #0177572  ;MMU OFF
3445 016620 005037 001042    CLR  #0FLAG    ;CLEAR MMU ABURT FLAG
3446 016624 012704 017200    MOV  #TMM16,R4 ;INIT R4
3447 016630 004767 162452    JSR  PC,MMU    ;INIT MMU
3448 016634 012737 000037 172516  MOV  #37,#0172516 ;ENABLE CSM INSTRUCTION
3449 016642 005037 177776    CLR  #0177776  ;SET PS TO KER MODE
3450 016646 013746 000010    MOV  #010, (SP) ;SAVE VECTORS
3451 016652 013746 000014    MOV  #014, -(SP)
3452 016656 013746 000016    MOV  #016, -(SP)

```

```

3453 016662 012737 017054 000010      MOV      0TMM16B,0010      ; SETUP NEW VECTORS
3454 016670 012737 000137 000014      MOV      0137,0014
3455 016676 012737 017214 000016      MOV      0TMM16A,0016
3456 016704 007014                      .WORD    7014
3457 016706 104000                      ERROR
3458 016710 000267                      .WORD    267
3459 016712 001213                      .WORD    MMUERR
3460
3461 016714 012737 017110 000010  TSM16A: MOV      0TMM16C,0010      ; GO TO ERROR IF NOT TRAPPED
3462 016722 012737 000027 172516      MOV      027,00172516      ; SETUP NEW VECTOR
3463 016730 012737 140000 177776      MOV      0140000,00177776  ; DISABLE CSM INSTRUCTION
3464 016736 007014                      .WORD    7014
3465 016740 104000                      ERROR
3466 016742 000270                      .WORD    270
3467 016744 001213                      .WORD    MMUERR
3468
3469 016746 012737 017144 000010  TSM16B: MOV      0TMM16D,0010  ; GO TO ERROR IF NOT TRAPPED
3470 016754 012737 000027 172516      MOV      027,00172516      ; SETUP NEW VECTOR
3471 016762 005037 177776      CLR      00177776
3472 016766 007014                      .WORD    7014
3473 016770 104000                      ERROR
3474 016772 000271                      .WORD    271
3475 016774 001213                      .WORD    MMUERR
3476
3477 016776 012737 000037 172516  TSM16C: MOV      037,00172516  ; GO TO ERROR IF NOT TRAPPED
3478 017004 012737 040000 177776      MOV      040000,00177776  ; ENABLE CSM INSTRUCTION
3479 017012 012706 000700                      MOV      0700,R6
3480 017016 012737 140000 177776      MOV      0140000,00177776  ; SET PS TO SUP MODE
3481 017024 012706 000600                      MOV      0600,R6
3482 017030 012737 000014 000010      MOV      014,0010
3483 017036 000277                      SCC
3484 017040 007024                      .WORD    7024
3485 017042                      TSM16D:
3486 017042 104000                      ERROR
3487 017044 000272                      .WORD    272
3488 017046 001213                      .WORD    MMUERR
3489
3490 017050 000167 000634                      JMP      TM16A
3491
3492
3493 017054 042737 007777 177776  TMM16B: BIC      07777,00177776  ; CLEAR UNWANTED BITS
3494 017062 022737 000000 177776      CMP      00,00177776
3495 017070 001403                      BEQ      1$
3496 017072 104000                      ERROR
3497 017074 000273                      .WORD    273
3498 017076 001213                      .WORD    MMUERR
3499
3500 017100 005726                      1$: TST      (SP),
3501 017102 005726                      TST      (SP),
3502 017104 000167 177604                      JMP      TSM16A
3503 017110 042737 007777 177776  TMM16C: BIC      07777,00177776  ; CLEAN UP STACK
3504 017116 022737 030000 177776      CMP      030000,00177776  ; CONTINUE TESTING
3505 017124 001403                      BEQ      1$
3506 017126 104000                      ERROR
3507 017130 000274                      .WORD    274
3508 017132 001213                      .WORD    MMUERR

```

```

3509
3510 017134 005726          1$:   TST      (SP)+      ;NO GO TO ERROR
3511 017136 005726          TST      (SP)+      ;CLEAN UP STACK
3512 017140 000167 177602          JMP      TSM16B      ;
3513 017144 042737 007777 177776 TMM16D: BIC      07777,00177776 ;CONTINUE TESTING
3514 017152 022737 000000 177776   CMP      00,00177776 ;CLEAR UNWANTED BITS
3515 017160 001403          BEQ      1$          ;IS PS CORRECT
3516 017162 104000          ERROR          ;YES GO ON
3517 017164 000275          .WORD    275        ;ALL ERRORS TO TRAP TO EMT VECTOR
3518 017166 001213          .WORD    MMUERR     ;UNIQUE ERROR NUMBER
3519                                .WORD          ;ADDRESS OF ERROR MESSAGE
3520 017170 005726          1$:   TST      (SP)+      ;NO GO TO ERROR
3521 017172 005726          TST      (SP)+      ;CLEAN UP STACK
3522 017174 000167 177576          JMP      TSM16C      ;
3523 017200 156430          TMM16E: .WORD    156430 ;CONTINUE TESTING
3524 017202          TMM16F:          ;TEST LOCATION
3525 017202 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3526 017204 000276          .WORD    276        ;UNIQUE ERROR NUMBER
3527 017206 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3528                                ;GO TO ERROR IF DIDN'T ABORT
3529 017210 000167 000474          JMP      TM16A
3530 017214 022737 070017 177776 TMM16A: CMP      070017,00177776 ;IS PS CORRECT
3531 017222 001403          BEQ      1$          ;YES GO ON
3532 017224 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3533 017226 000277          .WORD    277        ;UNIQUE ERROR NUMBER
3534 017230 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3535                                ;NO GO TO ERROR
3536 017232 020627 000572          1$:   CMP      R6,0572   ;IS SP CORRECT
3537 017236 001403          BEQ      2$          ;YES GO ON
3538 017240 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3539 017242 000300          .WORD    300        ;UNIQUE ERROR NUMBER
3540 017244 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3541                                ;NO GO TO ERROR
3542 017246 020427 017202          2$:   CMP      R4,0TMM16E+2 ;IS R4 CORRECT
3543 017252 001403          BEQ      3$          ;YES GO ON
3544 017254 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3545 017256 000301          .WORD    301        ;UNIQUE ERROR NUMBER
3546 017260 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3547                                ;NO GO TO ERROR
3548 017262 023727 017200 156430 3$:   CMP      00TMM16E,0156430 ;IS TEST LOCATION OK
3549 017270 001403          BEQ      4$          ;YES GO ON
3550 017272 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3551 017274 000302          .WORD    302        ;UNIQUE ERROR NUMBER
3552 017276 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3553                                ;NO GO TO ERROR
3554 017300 022627 156430          4$:   CMP      (SP)+,0156430 ;IS STACK CORRECT
3555 017304 001403          BEQ      5$          ;YES GO ON
3556 017306 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3557 017310 000303          .WORD    303        ;UNIQUE ERROR NUMBER
3558 017312 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE
3559                                ;NO GO TO ERROR
3560 017314 022627 017042          5$:   CMP      (SP)+,0TSM16D ;IS STACK CORRECT
3561 017320 001403          BEQ      6$          ;YES GO ON
3562 017322 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3563 017324 000304          .WORD    304        ;UNIQUE ERROR NUMBER
3564 017326 001213          .WORD    MMUERR     ;ADDRESS OF ERROR MESSAGE

```



```

3565
3566 017330 022627 140000      6$:  CMP      (SP)+, #140000      ;NO GO TO ERROR
3567 017334 001403              BEQ      7$                ;IS STACK CORRECT
3568 017336 104000              ERROR                      ;YES GO ON
3569 017340 000305              .WORD   305                ;ALL ERRORS TO TRAP TO EMT VECTOR
3570 017342 001213              .WORD   MMUERR             ;UNIQUE ERROR NUMBER
3571                                     ;ADDRESS OF ERROR MESSAGE
3572 017344 012706 000700      7$:  MOV      #700, R6          ;NO GO TO ERROR
3573 017350 012737 140000 177776  MOV      #140000, #177776 ;RESTORE SUP SP
3574 017356 020627 000600      CMP      R6, #600         ;SET PS TO USER MODE
3575 017362 001403              BEQ      8$                ;IS USER SP CORRECT
3576 017364 104000              ERROR                      ;YES GO ON
3577 017366 000306              .WORD   306                ;ALL ERRORS TO TRAP TO EMT VECTOR
3578 017370 001213              .WORD   MMUERR             ;UNIQUE ERROR NUMBER
3579                                     ;ADDRESS OF ERROR MESSAGE
3580 017372 012767 077400 152600 8$:  MOV      #77400, SIPDRO   ;NO GO TO ERROR
3581 017400 012737 017202 000016  MOV      #TMM16F, #16     ;SETUP SIPDRO TO ABORT
3582 017406 012737 000001 001042  MOV      #1, #FLAG        ;SETUP VECTOR
3583 017414 012737 000001 177572  MOV      #1, #177572      ;SETUP FLAG FOR AN ABORT
3584 017422 010701              MOV      R7, R1           ;TURN MMU ON
3585 017424 007014              .WORD   7014              ;SAVE OLD PC
3586 017426 022737 000000 001042  CMP      #0, #FLAG        ;TEST INSTRUCTION
3587 017434 001403              BEQ      9$                ;DID AN ABORT OCCUR
3588 017436 104000              ERROR                      ;YES GO ON
3589 017440 000307              .WORD   307                ;ALL ERRORS TO TRAP TO EMT VECTOR
3590 017442 001213              .WORD   MMUERR             ;UNIQUE ERROR NUMBER
3591                                     ;ADDRESS OF ERROR MESSAGE
3592 017444 023701 001076      9$:  CMP      #SAVMR2, R1     ;NO GO TO ERROR
3593 017450 001403              BEQ      10$               ;IS MM2 CORRECT
3594 017452 104000              ERROR                      ;YES GO ON
3595 017454 000310              .WORD   310                ;ALL ERRORS TO TRAP TO EMT VECTOR
3596 017456 001213              .WORD   MMUERR             ;UNIQUE ERROR NUMBER
3597                                     ;ADDRESS OF ERROR MESSAGE
3598 017460 023727 001072 100041 10$:  CMP      #SAVMR0, #100041 ;NO GO TO ERROR
3599 017466 001403              BEQ      11$               ;IS MM0 CORRECT
3600 017470 104000              ERROR                      ;YES GO ON
3601 017472 000311              .WORD   311                ;ALL ERRORS TO TRAP TO EMT VECTOR
3602 017474 001213              .WORD   MMUERR             ;UNIQUE ERROR NUMBER
3603                                     ;ADDRESS OF ERROR MESSAGE
3604 017476 012737 000037 172516 11$:  MOV      #37, #172516     ;NO GO TO ERROR
3605 017504 012737 040000 177776  MOV      #40000, #177776 ;ENABLE CSM
3606 017512 012706 000700      MOV      #700, R6         ;SET PSW TO SUP
3607 017516 012737 140000 177776  MOV      #140000, #177776 ;SETUP SUP SP
3608 017524 012706 000600      MOV      #600, R6         ;SET PSW TO USE
3609 017530 012737 000014 000010  MOV      #14, #10         ;SETUP USE SP
3610 017536 012737 017564 000016  MOV      #TS16, #16       ;SETUP NEW VECTOR
3611 017544 000277              SCC                       ;SETUP NEW VECTOR
3612 017546 007027              .WORD   7027              ;SET ALL CC BITS
3613 017550 045712              .WORD   4571?            ;TEST INSTRUCTION
3614 017552                                     TS16A:
3615 017552 104000              ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
3616 017554 000312              .WORD   312                ;UNIQUE ERROR NUMBER
3617 017556 001213              .WORD   MMUERR             ;ADDRESS OF ERROR MESSAGE
3618                                     ;GO TO ERROR IF DIDN'T TRAP
3619 017560 00016? 000124      JMP      TM16A
3620 017564 022737 070017 177776 TS16:  CMP      #70017, #177776 ;IS PSW CORRECT

```

```

3668 .MCALL IDMSG,ENDPAS
3669 .SBTTL END OF PASS ROUTINE
3670
3671 ;*****
3672 ;*INCREMENT THE PASS NUMBER ($PASS)
3673 ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
3674 ;*IF THERES A MONITOR GO TO IT
3675 ;*IF THERE ISN'T JUMP TO RESTART
3676
3677 $EOP:
3678 017740 005767 161042 TST $PASS ;ONLY TYPE MESSAGE AT END OF FIRST PASS
3679 017744 001002 BNE SKIPID ;IF >0 THEN SKIP THE ID MESSAGE
3680 017746 104401 020046 TYPE ,MSG1 ;ELSE TYPE THE ID MESSAGE
3681 017752
3682 017752 005267 161030 SKIPID: INC $PASS ;;INCREMENT THE PASS NUMBER
3683 017756 042767 100000 161022 BIC #100000,$PASS ;;DON'T ALLOW A NEG. NUMBER
3684 017764 005327 DEC (PC)+ ;;LOOP?
3685 017766 000001 $EOPCT: .WORD 1
3686 017770 003022 BGT $DOAGN ;;YES
3687 017772 012737 MOV (PC)+,@(PC)+ ;;RESTORE COUNTER
3688 017774 000001 $ENDCT: .WORD 1
3689 017776 017766 $EOPCT
3690 020000 104401 020125 TYPE ,MSG2
3691 020004 016746 160776 MOV $PASS,-(SP) ;;SAVE $PASS FOR TYPEOUT
3692 020010 104405 TYPDS ;;GO TYPE--DECIMAL ASCII WITH SIGN
3693 020012 104401 020042 TYPE , $ENULL
3694 020016 013700 000042 $GET42: MOV #42,RO ;;GET MONITOR ADDRESS
3695 020022 001405 BEQ $DOAGN ;;BRANCH IF NO MONITOR
3696 020024 000005 RESET ;;CLEAR THE WORLD
3697 020026 004710 $ENDAD: JSR PC,(RO) ;;GO TO MONITOR
3698 020030 000240 NOP ;;SAVE ROOM
3699 020032 000240 NOP ;;FOR
3700 020034 000240 NOP ;;ACT11
3701 020036
3702 020036 000137 $DOAGN: JMP @(PC)+ ;;RETURN
3703 020040 002066 $RTNAD: .WORD RESTART
3704 020042 377 000 $ENULL: .BYTE -1,-1,0 ;;NULL CHARACTER STRING
3705 020046 .EVEN
3706 020046 005015 055103 042113 MSG1: .ASCIZ <CR><LF>/CZKDK-B-0 KDJ11 MEMORY MANAGEMENT DIAGNOSTIC/
3707 020054 026513 026502 020060
3708 020062 042113 030512 020061
3709 020070 042515 047515 054522
3710 020076 046440 047101 043501
3711 020104 046505 047105 020124
3712 020112 044504 043501 047516
3713 020120 052123 041511 000
3714 020125 015 041412 045532 MSG2: .ASCIZ <CR><LF>/CZKDKB END PASS #/
3715 020132 045504 020102 047105
3716 020140 020104 040520 051523
3717 020146 021440 000
3718 020152
3719 .EVEN
3720 .SBTTL TYPE ROUTINE
3721 ;*****
3722 ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
3723 ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.

```

```

3724 ;*NOTE1:          $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
3725 ;*NOTE2:          $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
3726 ;*NOTE3:          $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
3727 ;*
3728 ;*CALL:
3729 ;*1) USING A TRAP INSTRUCTION
3730 ;*      TYPE      ,MESADR      ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
3731 ;*OR
3732 ;*      TYPE
3733 ;*      MESADR
3734 ;*
3735
3736 020152 105767 000343 $TYPE: TSTB      $TPFLG      ;; IS THERE A TERMINAL?
3737 020156 100002      BPL        1$          ;; BR IF YES
3738 020160 000000      HALT       ;; HALT HERE IF NO TERMINAL
3739 020162 000430      BR         3$          ;; LEAVE
3740 020164 010046      1$:      MOV      RO, -(SP)      ;; SAVE RO
3741 020166 017600      MOV      @2(SP),RO      ;; GET ADDRESS OF ASCIZ STRING
3742 020172 122767 000001 160620 CMPB     @APTENV,$ENV      ;; RUNNING IN APT MODE
3743 020200 001011      BNE      62$          ;; NO,GO CHECK FOR APT CONSOLE
3744 020202 132767 000100 160611 BITB     @APTSPool,$ENVM   ;; SPOOL MESSAGE TO APT
3745 020210 001405      BEQ      62$          ;; NO,GO CHECK FOR CONSOLE
3746 020212 010067 000004      MOV      RO,61$          ;; SETUP MESSAGE ADDRESS FOR APT
3747 020216 004767 001622      JSR     PC,$ATY3        ;; SPOOL MESSAGE TO APT
3748 020222 000000      61$:     .WORD     0          ;; MESSAGE ADDRESS
3749 020224 132767 000040 160567 62$:     BITB     @APTCSUP,$ENVM   ;; APT CONSOLE SUPPRESSED
3750 020232 001003      BNE      60$          ;; YES,SKIP TYPE OUT
3751 020234 112046      2$:      MOVVB   (RO)+,-(SP)      ;; PUSH CHARACTER TO BE TYPED ONTO STACK
3752 020236 001005      BNE      4$          ;; BR IF IT ISN'T THE TERMINATOR
3753 020240 005726      TST     (SP)+          ;; IF TERMINATOR POP IT OFF THE STACK
3754 020242 012600      60$:     MOV      (SP)+,RO      ;; RESTORE RO
3755 020244 062716 000002      3$:      ADD      @2,(SP)      ;; ADJUST RETURN PC
3756 020250 000002      RTI
3757 020252 122716 000011      4$:      CMPB     @HT,(SP)      ;; BRANCH IF <HT>
3758 020256 001430      BEQ      8$          ;; BRANCH IF NOT <CR LF>
3759 020260 122716 000200      CMPB     @CRLF,(SP)
3760 020264 001006      BNE      5$          ;; POP <CR><LF> EQUIV
3761 020266 005726      TST     (SP)+          ;; TYPE A CR AND LF
3762 020270 104401      TYPE
3763 020272 001277      $CRLF
3764 020274 105067 000202      CLRB     $CHARCNT      ;; CLEAR CHARACTER COUNT
3765 020300 000755      BR        2$          ;; GET NEXT CHARACTER
3766 020302 004767 000056      5$:      JSR     PC,$TYPEC      ;; GO TYPE THIS CHARACTER
3767 020306 126726 000206      6$:      CMPB     $FILLC,(SP)+  ;; IS IT TIME FOR FILLER CHARS.?
3768 020312 001350      BNE      2$          ;; IF NO GO GET NEXT CHAR.
3769 020314 016746 000176      MOV      $NULL,-(SP)   ;; GET # OF FILLER CHARS. NEEDED
3770 ;*AND THE NULL CHAR.
3771 020320 105366 000001      7$:      DECB     1(SP)        ;; DOES A NULL NEED TO BE TYPED?
3772 020324 002770      BLT     6$          ;; BR IF NO -GO POP THE NULL OFF OF STACK
3773 020326 004767 000032      JSR     PC,$TYPEC      ;; GO TYPE A NULL
3774 020332 105367 000144      DECB     $CHARCNT      ;; DO NOT COUNT AS A COUNT
3775 020336 000770      BR        7$          ;; LOOP
3776
3777 ;HORIZONTAL TAB PROCESSOR
3778
3779 020340 112716 000040      8$:      MOVVB   @' ,(SP)      ;; REPLACE TAB WITH SPACE

```

```

3780 020344 004767 000014 9$: JSR PC,$TYPEC ;;TYPE A SPACE
3781 020350 132767 000007 000124 BITB #7,$CHARCNT ;;BRANCH IF NOT AT
3782 020356 001372 BNE 9$ ;;TAB STOP
3783 020360 005726 TST (SP)+ ;;POP SPACE OFF STACK
3784 020362 000724 BR 2$ ;;GET NEXT CHARACTER
3785 020364 $TYPEC:
3786 020364 105777 000116 TSTB @TKS ;;CHAR IN KYBD BUFFER? ;MJD001
3787 020370 100022 BPL 10$ ;;BR IF NOT ;MJD001
3788 020372 017746 000112 MOV @TKB,-(SP) ;;GET CHAR ;MJD001
3789 020376 042716 177600 BIC #177600,(SP) ;;STRIP EXTRANEIOUS BITS ;MJD001
3790 020402 122716 000023 CMPB #XOFF,(SP) ;;WAS CHAR XOFF ;MJD001
3791 020406 001012 BNE 102$ ;;BR IF NOT ;MJD001
3792 020410 101$:
3793 020410 105777 000072 TSTB @TKS ;;WAIT FOR CHAR ;MJD001
3794 020414 100375 BPL 101$ ;MJD001
3795 020416 117716 000066 MOVB @TKB,(SP) ;;GET CHAR ;MJD001
3796 020422 042716 177600 BIC #177600,(SP) ;;STRIP IT ;MJD001
3797 020426 122716 000021 CMPB #XON,(SP) ;;WAS IT XON? ;MJD001
3798 020432 001366 BNE 101$ ;;BR IF NOT ;MJD001
3799 020434 102$:
3800 020434 005726 TST (SP)+ ;;FIX STACK ;MJD001
3801 020436 10$:
3802 020436 105777 000050 TSTB @TPS ;;WAIT UNTIL PRINTER IS READY ;MJD001
3803 020442 100375 BPL 10$ ;MJD001
3804 020444 116677 000002 00004? MOVB 2(SP),@TPB ;;LOAD CHAR TO BE TYPED INTO DATA REG.
3805 020452 122766 000015 000002 CMPB #CR,2(SP) ;;IS CHARACTER A CARRIAGE RETURN?
3806 020460 001003 BNE 1$ ;;BRANCH IF NO
3807 020462 105067 000014 CLRB $CHARCNT ;;YES--CLEAR CHARACTER COUNT
3808 020466 000406 BR $TYPEX ;;EXIT
3809 020470 122766 000012 000002 1$: CMPB #LF,2(SP) ;;IS CHARACTER A LINE FEED?
3810 020476 001402 BR $TYPEX ;;BRANCH IF YES
3811 020500 105227 INCB (PC)+ ;;COUNT THE CHARACTER
3812 020502 000000 $CHARCNT: .WORD 0 ;;CHARACTER COUNT STORAGE
3813 020504 000207 $TYPEX: RTS PC
3814
3815 020506 177560 $TKS: .WORD 177560 ;;TTY KDB STATUS ;MJD001
3816 020510 177562 $TKB: .WORD 177562 ;;TTY KBD BUFFER ;MJD001
3817 020512 177564 $TPS: .WORD 177564 ;;TTY PRINTER STATUS REG. ADDRESS
3818 020514 177566 $TPB: .WORD 177566 ;;TTY PRINTER BUFFER REG. ADDRESS
3819 020516 000 $NULL: .BYTE 0 ;;CONTAINS NULL CHARACTER FOR FILLS
3820 020517 002 $FILLS: .BYTE 2 ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
3821 020520 012 $FILLC: .BYTE 12 ;;INSERT FILL CHARS. AFTER A "LINE FEED"
3822 020521 000 $TPFLG: .BYTE 0 ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
3823 020522 077 $QUES: .ASCII "?" ;;QUESTION MARK
3824 020523 012 000 $LF: .ASCIIZ <12> ;;LINEFEED
3825 020526 .EVEN
3826
3827 .SBTTL CONVERT BINARY TO DECIMAL AND TYPE ROUTINE
3828
3829 *****
3830 ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
3831 ;*SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
3832 ;*NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
3833 ;*BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
3834 ;*REPLACED WITH SPACES.
3835 ;*CALL:
;* MOV NUM,-(SP) ;;PUT THE BINARY NUMBER ON THE STACK

```

```

3892 020742 000004 $DBLK: .BLKW 4
3893 .SBTTL BINARY TO OCTAL (ASCII) AND TYPE
3894
3895 ;*****
3896 ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
3897 ;*OCTAL (ASCII) NUMBER AND TYPE IT.
3898 ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
3899 ;*CALL:
3900 ;*      MOV      NUM, -(SP)      ;;NUMBER TO BE TYPED
3901 ;*      TYPOS    ;;CALL FOR TYPEOUT
3902 ;*      .BYTE   N                ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
3903 ;*      .BYTE   M                ;;M=1 OR 0
3904 ;*
3905 ;*                               ;;1=TYPE LEADING ZEROS
3906 ;*                               ;;0=SUPPRESS LEADING ZEROS
3907 ;*$TYPON---ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
3908 ;*$TYPOS OR $TYPOC
3909 ;*CALL:
3910 ;*      MOV      NUM, -(SP)      ;;NUMBER TO BE TYPED
3911 ;*      TYPON    ;;CALL FOR TYPEOUT
3912 ;*
3913 ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
3914 ;*CALL:
3915 ;*      MOV      NUM, -(SP)      ;;NUMBER TO BE TYPED
3916 ;*      TYPOC    ;;CALL FOR TYPEOUT
3917
3918 020752 017646 000000 $TYPOS: MOV      @ (SP), -(SP)      ;;PICKUP THE MODE
3919 020756 116667 000001 000211 MOVVB   1 (SP), $OFILL      ;;LOAD ZERO FILL SWITCH
3920 020764 112667 000207 MOVVB   (SP)+, $OMODE+1    ;;NUMBER OF DIGITS TO TYPE
3921 020770 062716 000002 ADD      #2, (SP)        ;;ADJUST RETURN ADDRESS
3922 020774 000406 BR      $TYPON
3923 020776 112767 000001 000171 $TYPOC: MOVVB   #1, $OFILL      ;;SET THE ZERO FILL SWITCH
3924 021004 112767 000006 000165 MOVVB   #6, $OMODE+1      ;;SET FOR SIX(6) DIGITS
3925 021012 112767 000005 000154 $TYPON: MOVVB   #5, $OCNT      ;;SET THE ITERATION COUNT
3926 021020 010346 MOV      R3, -(SP)      ;;SAVE R3
3927 021022 010446 MOV      R4, -(SP)      ;;SAVE R4
3928 021024 010546 MOV      R5, -(SP)      ;;SAVE R5
3929 021026 116704 000145 MOVVB   $OMODE+1, R4      ;;GET THE NUMBER OF DIGITS TO TYPE
3930 021032 005404 NEG      R4
3931 021034 062704 000006 ADD      #6, R4          ;;SUBTRACT IT FOR MAX. ALLOWED
3932 021040 110467 000132 MOVVB   R4, $OMODE      ;;SAVE IT FOR USE
3933 021044 116704 000125 MOVVB   $OFILL, R4      ;;GET THE ZERO FILL SWITCH
3934 021050 016605 000012 MOV      12 (SP), R5    ;;PICKUP THE INPUT NUMBER
3935 021054 005003 CLR      R3            ;;CLEAR THE OUTPUT WORD
3936 021056 006105 1$: ROL      R5            ;;ROTATE MSB INTO "C"
3937 021060 000404 BR      3$           ;;GO DO MSB
3938 021062 006105 2$: ROL      R5            ;;FORM THIS DIGIT
3939 021064 006105 ROL      R5
3940 021066 006105 ROL      R5
3941 021070 010503 MOV      R5, R3
3942 021072 006103 3$: ROL      R3            ;;GET LSB OF THIS DIGIT
3943 021074 105367 000076 DECB   $OMODE          ;;TYPE THIS DIGIT?
3944 021100 100016 BPL      7$           ;;BR IF NO
3945 021102 042703 177770 BIC   #177770, R3     ;;GET RID OF JUNK
3946 021106 001002 BNE     4$           ;;TEST FOR 0
3947 021110 005704 TST     R4            ;;SUPPRESS THIS 0?

```

```

3948 021112 001403          BEQ      5$          ;;BR IF YES
3949 021114 005204          4$: INC      R4          ;;DON'T SUPPRESS ANYMORE 0'S
3950 021116 052703 000060  BIS      #'0,R3      ;;MAKE THIS DIGIT ASCII
3951 021122 052703 000040  5$: BIS      #' ,R3      ;;MAKE ASCII IF NOT ALREADY
3952 021126 110367 000040  MOV      R3,8$        ;;SAVE FOR TYPING
3953 021132 104401 021172  TYPE      ,8$        ;;GO TYPE THIS DIGIT
3954 021136 105367 000032  7$: DECB    $OCNT      ;;COUNT BY 1
3955 021142 003347          BGT      2$          ;;BR IF MORE TO DO
3956 021144 002402          BIT      6$          ;;BR IF DONE
3957 021146 005204          INC      R4          ;;INSURE LAST DIGIT ISN'T A BLANK
3958 021150 000744          BR       2$          ;;GO DO THE LAST DIGIT
3959 021152 012605          6$: MOV     (SP)+,R5    ;;RESTORE R5
3960 021154 012604          MOV     (SP)+,R4    ;;RESTORE R4
3961 021156 012603          MOV     (SP)+,R3    ;;RESTORE R3
3962 021160 016666 000002 000004  MOV     2(SP),4(SP)  ;;SET THE STACK FOR RETURNING
3963 021166 012616          MOV     (SP)+,(SP)
3964 021170 000002          RTI          ;;RETURN
3965 021172          8$: .BYTE   0          ;;STORAGE FOR ASCII DIGIT
3966 021173          .BYTE   0          ;;TERMINATOR FOR TYPE ROUTINE
3967 021174          .BYTE   0          ;;OCTAL DIGIT COUNTER
3968 021175          .BYTE   0          ;;ZERO FILL SWITCH
3969 021176 000000          $OMODE: .WORD  0     ;;NUMBER OF DIGITS TO TYPE
3970          .SBTTL  ITY INPUT ROUTINE
3971
3972          ;;*****
3973          .ENABL  LSB
3974
3975          ;;*****
3976          ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
3977          ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
3978          ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
3979          ;*WHEN OPERATING IN ITY FLAG MODE.
3980 021200 022767 000176 157640  $CKSWR: CMP     $SWREG,SWR      ;;IS THE SOFT-SWR SELECTED?
3981 021206 001074          BNE     15$          ;;BRANCH IF NO
3982 021210 105777 177272          TSTB   @TKS          ;;CHAR THERE?
3983 021214 100071          BPL     15$          ;;IF NO, DON'T WAIT AROUND
3984 021216 117746 177266          MOV     @TKB,-(SP)      ;;SAVE THE CHAR
3985 021222 042716 177600          BIC     @C177,(SP)     ;;STRIP-OFF THE ASCII
3986 021226 022726 000007          CMP     @7,(SP)+       ;;IS IT A CONTROL G?
3987 021232 001062          BNE     15$          ;;NO, RETURN TO USER
3988 021234 126727 000514 000001  CMP     $AUTOB,#1      ;;ARE WE RUNNING IN AUTO-MODE?
3989 021242 001456          BEQ     15$          ;;BRANCH IF YES
3990
3991 021244 104401 021725          TYPE   , $CNTLG      ;;ECHO THE CONTROL-G (+G)
3992 021250 104401 021732          $GTSWR: TYPE   , $MSWR      ;;TYPE CURRENT CONTENTS
3993 021254 016746 156716          MOV     SWREG,-(SP)    ;;SAVE SWREG FOR TYPEOUT
3994 021260 104402          TYPOC          ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
3995 021262 104401 021743          TYPE   , $MNEW      ;;PROMPT FOR NEW SWR
3996 021266 005046          19$: CLR     -(SP)     ;;CLEAR COUNTER
3997 021270 005046          CLR     -(SP)     ;;THE NEW SWR
3998 021272 105777 177210          7$: TSTB   @TKS          ;;CHAR THERE?
3999 021276 100375          BPL     7$          ;;IF NOT TRY AGAIN
4000
4001 021300 117746 177204          MOV     @TKB,-(SP)    ;;PICK UP CHAR
4002 021304 042716 177600          BIC     @C177,(SP)    ;;MAKE IT 7-BIT ASCII
4003

```

```

4004
4005
4006 021310 021627 000025      9$:    CMP      (SP),#25      ;; IS IT A CONTROL-U?
4007 021314 001005                BNE      10$      ;; BRANCH IF NOT
4008 021316 104401 021720        TYPE     , $CNTLU    ;; YES, ECHO CONTROL-U (^U)
4009 021322 062706 000006      20$:    ADD      #6,SP      ;; IGNORE PREVIOUS INPUT
4010 021326 000757                BR       19$      ;; LET'S TRY IT AGAIN
4011
4012
4013 021330 021627 000015      10$:    CMP      (SP),#15     ;; IS IT A <CR>?
4014 021334 001022                BNE      16$      ;; BRANCH IF NO
4015 021336 005766 000004        TST      4(SP)      ;; YES, IS IT THE FIRST CHAR?
4016 021342 001403                BEQ      11$      ;; BRANCH IF YES
4017 021344 016677 000002 157474    MOV      2(SP),@SWR  ;; SAVE NEW SWR
4018 021352 062706 000006      11$:    ADD      #6,SP      ;; CLEAR UP STACK
4019 021356 104401 001277      14$:    TYPE     , $CRLF    ;; ECHO <CR> AND <LF>
4020 021362 126727 000367 000001    CMPB    $INTAG,#1  ;; RE-ENABLE TTY KBD INTERRUPTS?
4021 021370 001003                BNE      15$      ;; BRANCH IF NOT
4022 021372 012777 000100 177106    MOV      #100,@TKS  ;; RE-ENABLE TTY KBD INTERRUPTS
4023 021400 000002                RTI                     ;; RETURN
4024 021402 004767 176756      16$:    JSR      PC,$TYPEC  ;; ECHO CHAR
4025 021406 021627 000060        CMP      (SP),#60   ;; CHAR < 0?
4026 021412 002420                BLT      18$      ;; BRANCH IF YES
4027 021414 021627 000067        CMP      (SP),#67   ;; CHAR > 7?
4028 021420 003015                BGT      12$      ;; BRANCH IF YES
4029 021422 042726 000060        BIC      #60,(SP)+  ;; STRIP-OFF ASCII
4030 021426 005766 000002        TST      2(SP)      ;; IS THIS THE FIRST CHAR
4031 021432 001403                BEQ      17$      ;; BRANCH IF YES
4032 021434 006316                ASL      (SP)        ;; NO, SHIFT PRESENT
4033 021436 006316                ASL      (SP)        ;; CHAR OVER TO MAKE
4034 021440 006316                ASL      (SP)        ;; ROOM FOR NEW ONE.
4035 021442 005266 000002      17$:    INC      2(SP)      ;; KEEP COUNT OF CHAR
4036 021446 056616 177776        BIS      -2(SP),(SP) ;; SET IN NEW CHAR
4037 021452 000707                BR       7$        ;; GET THE NEXT ONE
4038 021454 104401 020522      18$:    TYPE     , $QUES    ;; TYPE ?<CR><LF>
4039 021460 000720                BR       20$      ;; SIMULATE CONTROL-U
4040      .DSABL  LSB
4041
4042
4043
4044      ;;*****
4045      ;; THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
4046      ;; CALL:
4047      ;;      RDCHR                ;; INPUT A SINGLE CHARACTER FROM THE TTY
4048      ;;      RETURN HERE          ;; CHARACTER IS ON THE STACK
4049      ;;                          ;; WITH PARITY BIT STRIPPED OFF
4050
4051      $RDCHR: MOV      (SP),-(SP)  ;; PUSH DOWN THE PC
4052 021464 016666 000004 000002    MOV      4(SP),2(SP) ;; SAVE THE PS
4053 021472 105777 177010      1$:    TSTB    @TKS      ;; WAIT FOR
4054 021476 100375                BPL      1$        ;; A CHARACTER
4055 021500 117766 177004 000004    MOVB    @TKB,4(SP)  ;; READ THE TTY
4056 021506 042766 177600 000004    BIC      #C<177>,4(SP) ;; GET RID OF JUNK IF ANY
4057 021514 026627 000004 000023    CMP      4(SP),#23  ;; IS IT A CONTROL-S?
4058 021522 001013                BNE      3$        ;; BRANCH IF NO
4059 021524 105777 176756      2$:    TSTB    @TKS      ;; WAIT FOR A CHARACTER

```

```

4060 021530 100375          BPL      2#          ;;LOOP UNTIL ITS THERE.
4061 021532 117746 176752  MOVB    0#TKB,-(SP)  ;;GET CHARACTER
4062 021536 042716 177600  BIC     0#C177,(SP)  ;;MAKE IT 7-BIT ASCII
4063 021542 022627 000021  CMP     (SP)+,021    ;;IS IT A CONTROL-Q?
4064 021546 001366          BNE     2#          ;;IF NOT DISCARD IT
4065 021550 000750          BR      1#          ;;YES, RESUME
4066 021552 026627 000004 000140 3#;    CMP     4(SP),0140  ;;IS IT UPPER CASE?
4067 021560 002407          BLT     4#          ;;BRANCH IF YES
4068 021562 026627 000004 000175  CMP     4(SP),0175  ;;IS IT A SPECIAL CHAR?
4069 021570 003003          BGT     4#          ;;BRANCH IF YES
4070 021572 042766 000040 000004  BIC     040,4(SP)   ;;MAKE IT UPPER CASE
4071 021600 000002          4#;    RTI          ;;GO BACK TO USER
4072
4073 *****
4074 ;*THIS ROUTINE WILL INPUT A STRING FROM THE TTY
4075 ;*CALL:
4076 ;*      RDLIN          ;;INPUT A STRING FROM THE TTY
4077 ;*      RETURN HERE   ;;ADDRESS OF FIRST CHARACTER WILL BE ON THE STACK
4078 ;*                  ;;TERMINATOR WILL BE A BYTE OF ALL 0'S
4079
4079 021602 010346          $RDLIN: MOV     R3,-(SP)  ;;SAVE R3
4080 021604 012703 021710  1#;    MOV     0#TTYIN,R3  ;;GET ADDRESS
4081 021610 022703 021720  2#;    CMP     0#TTYIN+8,,R3  ;;BUFFER FULL?
4082 021614 101405          BLOS   4#          ;;BR IF YES
4083 021616 104410          RDCHR          ;;GO READ ONE CHARACTER FROM THE TTY
4084 021620 112613          MOVB   (SP)+,(R3)  ;;GET CHARACTER
4085 021622 122713 000177  10#;   CMPB   0177,(R3)  ;;IS IT A RUBOUT
4086 021626 001003          BNE   3#          ;;SKIP IF NOT
4087 021630 104401 020522  4#;    TYPE   ,QUES    ;;TYPE A '?'
4088 021634 000763          BR      1#          ;;CLEAR THE BUFFER AND LOOP
4089 021636 111367 000044  3#;    MOVB   (R3),9#    ;;ECHO THE CHARACTER
4090 021642 104401 021706          TYPE   ,9#
4091 021646 122723 000015          CMPB   015,(R3)+   ;;CHECK FOR RETURN
4092 021652 001356          BNE   2#          ;;LOOP IF NOT RETURN
4093 021654 105063 177777          CLRB  -1(R3)      ;;CLEAR RETURN (THE 15)
4094 021660 104401 020523          TYPE   ,LF        ;;TYPE A LINE FEED
4095 021664 012603          MOV    (SP)+,R3   ;;RESTORE R3
4096 021666 011646          MOV    (SP),-(SP) ;;ADJUST THE STACK AND PUT ADDRESS OF THE
4097 021670 016666 000004 000002  MOV    4(SP),2(SP) ;;FIRST ASCII CHARACTER ON IT
4098 021676 012766 021710 000004  MOV    0#TTYIN,4(SP)
4099 021704 000002          RTI          ;;RETURN
4100 021706          000          9#;    .BYTE   0          ;;STORAGE FOR ASCII CHAR. TO TYPE
4101 021707          000          .BYTE   0          ;;TERMINATOR
4102 021710 000010          $TTYIN: .BLKB  8.    ;;RESERVE 8 BYTES FOR TTY INPUT
4103 021720 052536 005015          000 $CNTLU: .ASCIZ  /LU/15/12/  ;;CONTROL "U"
4104 021725          136 006507 000012 $CNTLG: .ASCIZ  /LG/15/12/  ;;CONTROL "G"
4105 021732 005015 053523 020122 $MSWR:  .ASCIZ  <15><12>/SWR = /
4106 021740 020075          000
4107 021743          040 047040 053505 $MNEW:  .ASCIZ  / NEW = /
4108 021750 036440 000040
4109 021754          000
4110 021755          000
4111
4112
4113
4114
4115 *****
4115 ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
4115 ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS

```



```

4116 ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
4117 ;*GO TO THAT ROUTINE.
4118
4119 021756 010046 ;TRAP: MOV R0,-(SP) ;SAVE R0
4120 021760 016600 000002 MOV 2(SP),R0 ;GET TRAP ADDRESS
4121 021764 005740 TST -(R0) ;BACKUP BY 2
4122 021766 111000 MOVB (R0),R0 ;GET RIGHT BYTE OF TRAP
4123 021770 006300 ASI R0 ;POSITION FOR INDEXING
4124 021772 016000 022012 MOV $TRPAD(R0),R0 ;INDEX TO TABLE
4125 021776 000200 RTS R0 ;GO TO ROUTINE
4126
4127
4128 ;THIS IS USE TO HANDLE THE "GETPRI" MACRO
4129
4130 022000 011646 ;TRAP2: MOV (SP),-(SP) ;MOVE THE PC DOWN
4131 022002 016666 000004 000002 MOV 4(SP),2(SP) ;MOVE THE PSW DOWN
4132 022010 000002 RTI ;RESTORE THE PSW
4133
4134 .MACRO SETTRAP A,B,MSG
4135 ;SET A,B,\<TRAP+$TRP>,\$TRP,<MSG>
4136 .NLIST
4137 $TRP=$TRP+1
4138 .LIST
4139 .ENDM SETTRAP
4140 .MACRO ;SET A,B,C,D,COMNT
4141 .IF EQ $TRP-1
4142 .SBTTL TRAP TABLE
4143
4144 ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
4145 ;*BY THE "TRAP" INSTRUCTION.
4146
4147 ; ROUTINE
4148 ; -----
4149 $TRPAD: .WORD $TRAP2
4150 .ENDC
4151 .IF NDF GNS,.NLIST
4152 A= C
4153 .IF NDF GNS,.LIST
4154 B ;CALL=A TRAP+D(C) COMNT
4155 .ENDM ;SET
4156 .MACRO TRMTRP
4157 $TERM=-$TRPAD
4158 .ENDM TRMTRP
4159 .SBTTL TRAP TABLE
4160
4161 ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
4162 ;*BY THE "TRAP" INSTRUCTION.
4163
4164 ; ROUTINE
4165 ; -----
4166 022012 022000 $TRPAD: .WORD $TRAP2
4167 022014 020152 $TYPE ;CALL=TYPE TRAP+1(104401) TTY TYPEOUT ROUTINE
4168 022016 020776 $TYPOC ;CALL=TYPOC TRAP+2(104402) TYPE OCTAL NUMBER (WITH LEADING ZEROS)
4169 022020 020752 $TYPOS ;CALL=TYPOS TRAP+3(104403) TYPE OCTAL NUMBER (NO LEADING ZEROS)
4170 022022 021012 $TYPON ;CALL=TYPON TRAP+4(104404) TYPE OCTAL NUMBER (AS PER LAST CALL)
4171 022024 020526 $TYPDS ;CALL=TYPDS TRAP+5(104405) TYPE DECIMAL NUMBER (WITH SIGN)

```

BEVENT*	177546	487#																			
BIT0	* 000001	308#																			
BIT00	= 000001	298#	308	3304																	
BIT01	* 000002	297#	307																		
BIT02	* 000004	296#	306																		
BIT03	* 000010	295#	305																		
BIT04	= 000020	294#	304	3009	3025																
BIT05	= 000040	293#	303	3311																	
BIT06	* 000100	292#	302																		
BIT07	= 000200	291#	301																		
BIT08	* 000400	290#	300	2970																	
BIT09	* 001000	289#	299																		
BIT1	* 000002	307#																			
BIT10	= 002000	288#																			
BIT11	= 004000	287#																			
BIT12	* 010000	286#																			
BIT13	= 020000	285#																			
BIT14	= 040000	284#																			
BIT15	* 100000	283#																			
BIT2	* 000004	306#																			
BIT3	* 000010	305#																			
BIT4	= 000020	304#																			
BIT5	* 000040	303#																			
BIT6	* 000100	302#																			
BIT7	* 000200	301#																			
BIT8	= 000400	300#																			
BIT9	= 001000	299#																			
BNO	012144	2382	2489#																		
BN1	012334	2393	2404	2552#																	
BOARDT	= 000001	1#	538	637	668																
BPTVEC	= 000014	315#																			
CACHET	= 000001	1#	185	188	206	538	623	640	653	659	668	765	809	810							
		3666	3718	3970																	
CCR	* 177746	481#	766*	822*																	
CKSWR	* 104407	4175#																			
COUNT	001040	610#																			
CPEREG	= 177766	484#	856*	861*	3085*	3305*	3311	3327*													
CPUTST	= 000001	1#	195	538	634	663	824	3706													
CR	= 000015	223#	653	659	663	668	670	673	3706	3714	3805	3817									
CRLF	= 000200	224#	3759	3817																	
DATRAM	= 000001	1#	810																		
DCOUNT	001054	617#																			
DDISP	= 177570	230#	614	782																	
DISPLA	001050	614#	782*	790*																	
DISPRE	000174	524#	790																		
DSWR	= 177570	229#	613	781																	
EMTVEC	= 000030	318#	770*	771*																	
ERRCNT	001044	612#	4241*																		
ERRMSG	001166 G	659#																			
ERRNUM	= 000321	494#	743	746#	849	852#	897	900#	903	906#	913	916#	926	929#							
		942	945#	957	960#	970	973#	985	986#	996	999#	1009	1012#	1021							
		1024#	1032	1035#	1044	1047#	1058	1061#	1065	1068#	1076	1079#	1083	1086#							
		1091	1094#	1098	1101#	1108	1111#	1115	1118#	1125	1128#	1132	1135#	1142							
		1145#	1149	1152#	1157	1160#	1184	1187#	1192	1195#	1200	1203#	1208	1211#							
		1217	1220#	1225	1228#	1242	1245#	1251	1254#	1296	1299#	1302	1305#	1310							
		1313#	1316	1319#	1324	1327#	1330	1333#	1338	1341#	1344	1347#	1353	1356#							

KDPDR2*	172324	4390												
KDPDR3*	172326	4400												
KDPDR4*	172330	4410												
KDPDR5*	172332	4420												
KDPDR6*	172334	4430												
KDPDR7*	172336	4440												
KIPAR0*	172340	4480	3090*											
KIPAR1*	172342	4490												
KIPAR2*	172344	4500												
KIPAR3*	172346	4510												
KIPAR4*	172350	4520												
KIPAR5*	172352	4530												
KIPAR6*	172354	4540	2992*	3010*	3028*	3039*	3050*	3061*	3092*	3301*				
KIPAR7*	172356	4550												
KIPDR0*	172300	4260	3091*											
KIPDR1*	172302	4270												
KIPDR2*	172304	4280												
KIPDR3*	172306	4290												
KIPDR4*	172310	4300												
KIPDR5*	172312	4310												
KIPDR6*	172314	4320	3095*											
KIPDR7*	172316	4330												
LF	000012	2220	653	659	663	668	670	673	3706	3714	3809	3817		
LOOPIN	001060	6190												
MMTS10*	***** U	2592												
MMTS11*	***** U	2688												
MMTS12*	***** U	2745												
MMTS13*	***** U	2832												
MMTS14*	***** U	3076												
MMTS15*	***** U	3331												
MMTS16*	***** U	3438												
MMTS6A*	***** U	1574												
MMTS6B*	***** U	1712												
MMTS6C*	***** U	1885												
MMTS6D*	***** U	2035												
MMU	001306	6950	1589	1725	1898	2048	2198	2216	2321	2380	2603	2615	2628	2642
		2716	2754	2843	3104	3300	3339	3447						
MMUERR	001213	6630	745	851	899	905	915	928	944	959	972	985	998	1011
		1023	1034	1046	1060	1067	1078	1085	1093	1100	1110	1117	1127	1134
		1144	1151	1159	1186	1194	1202	1210	1219	1227	1244	1253	1298	1304
		1312	1318	1326	1332	1340	1346	1355	1361	1369	1375	1414	1420	1428
		1434	1442	1448	1456	1462	1471	1477	1485	1491	1510	1519	1528	1537
		1553	1559	1565	1571	1602	1610	1617	1623	1629	1636	1642	1653	1660
		1666	1673	1679	1688	1695	1701	1707	1738	1746	1753	1759	1765	1772
		1778	1789	1796	1802	1809	1815	1826	1833	1839	1846	1852	1861	1868
		1874	1880	1912	1920	1927	1933	1939	1946	1957	1964	1970	1977	1988
		1995	2001	2008	2017	2024	2030	2062	2070	2077	2083	2089	2096	2107
		2114	2120	2127	2138	2145	2151	2158	2167	2174	2180	2258	2265	2271
		2277	2290	2297	2303	2335	2342	2348	2354	2361	2426	2433	2440	2446
		2452	2662	2668	2674	2680	2727	2739	2813	2819	2825	2863	2885	3004
		3020	3035	3046	3057	3068	3120	3183	3189	3315	3320	3325	3393	3414
		3430	3459	3467	3475	3488	3498	3508	3518	3527	3534	3540	3546	3550
		3558	3564	3570	3578	3590	3596	3602	3617	3624	3630	3636	3642	3648
		3656												
MMUTRP	001460	7410	808											
MMUTST*	***** U	1	195	538	624	653	663	668	689	808	809	810	824	3700

		3970							
MMUTS1=	***** U	830							
MMUTS2=	***** U	855							
MMUTS3=	***** U	1165							
MMUTS4=	***** U	1258							
MMUTS5=	***** U	1496							
MMUTS6=	***** U	1540							
MMUTS7=	***** U	2185							
MMUTS8=	***** U	2308							
MMUTS9=	***** U	2368							
MMVEC =	000250	327*	808*						
MODE1	014334	2852	2945*						
MODE2	015770	3130	3247*						
MSER =	177744	482*							
MSG1	020046	3680	3706*						
MSG2	020125	3690	3714*						
MXVDEL =	000001	1*							
NOTOK	001542	751	757*						
NXMF IN	016232	3308	3322	3326*					
NXMTRP	015620	3088	3198*						
NXMTST=	***** U	3290							
OK	001520	750	752*						
OKAY7	011144	2268	2273*						
OKAY7A	011160	2274	2279*						
OKA7	011126	2262	2267*						
OK1	001546	756	758*						
OK7	011104	2255	2260*						
OPMSG2	001124	653*	809						
PAR	001430	700	701	704	705	707	708	726*	
PARAD1	014164	2845	2893*						
PARAD2	015652	3126	3208*						
PARRAM=	000001	1*							
PARVA1	014216	2846	2906*						
PARVA2	015704	3125	3221*						
PARVA3	016022	3108	3132	3260*					
PAR1	001432	727*	730						
PDR	001406	699	703	706	717*				
PDR1	001410	718*	721						
PHY1	014302	2850	2932*						
PIRQ =	177772	228*							
PIRQVE=	000240	322*							
PLFO	012072	2381	2468*						
PLF1	012264	2392	2403	2532*					
PRO =	000000	245*							
PR1 =	000040	246*							
PR2 =	000100	247*							
PR3 =	000140	248*							
PR4 =	000200	249*							
PR5 =	000240	250*							
PR6 =	000300	251*							
PR7 =	000340	252*							
PS =	177776	225*	226	3306*					
PSW =	177776	226*							
PWRVEC=	000024	317*							
QBUSEX=	000001	1*	633	3667					
RBUF =	177562	489*							

SR2	* 177576	3330							
SR3	* 172516	3340	2991*	3009*	3025*	3096*	3660*		
STACK	= 001000	2160	768						
START	001604	521	527	7650					
STBOT	= 001000	4990	1583	1586	3005	3072			
STKLMT	= 177774	2270							
SWR	001046	6130	781*	783	789*	803*	815	2970	3980
SWREG	000176	5250	789	815	3980	3993			4017*
SW0	= 000001	2800							
SW00	= 000001	2700	280						
SW01	= 000002	2690	279						
SW02	= 000004	2680	278						
SW03	= 000010	2670	277						
SW04	= 000020	2660	276						
SW05	= 000040	2650	275						
SW06	= 000100	2640	274						
SW07	= 000200	2630	273						
SW08	= 000400	2620	272						
SW09	= 001000	2610	271						
SW1	= 000002	2790							
SW10	= 002000	2600							
SW11	= 004000	2590							
SW12	= 010000	2580							
SW13	= 020000	2570							
SW14	= 040000	2560							
SW15	= 100000	2550							
SW2	= 000004	2780							
SW3	= 000010	2770							
SW4	= 000020	2760							
SW5	= 000040	2750							
SW6	= 000100	2740							
SW7	= 000200	2730							
SW8	= 000400	2720							
SW9	= 001000	2710							
TAGRAM	= 000001	10	810						
TRITVE	= 000014	3130							
TKVEC	= 000060	3200							
TMM16A	017214	3455	35300						
TMM16B	017054	3453	34930						
TMM16C	017110	3461	35030						
TMM16D	017144	3469	35130						
TMM16E	017200	3446	35230	3542	3548				
TMM16F	017202	35240	3581						
TM16A	017710	3490	3529	3619	3653	36580			
TPVEC	= 000064	3210							
TRAPVE	= 000034	3190	772*	773*					
TRTVEC	= 000014	3140							
TSMA	011214	2287	22920						
TSMB	011240	2294	22990						
TSMC	011254	2300	23050						
TSMMU1	002100	8300							
TSMMU2	002174	8550							
TSMMU3	003550	11650							
TSMMU4	004114	12580							
TSMMU5	005210	14960							
TSMMU6	005366	15400							

TSMPU7	010572	2185#																	
TSMMU8	011256	2308#																	
TSMMU9	011522	2368#																	
TSMM10	012450	2592#																	
TSMM11	013064	2688#																	
TSMM12	013336	2745#																	
TSMM13	013742	2832#																	
TSMM14	015064	3076#																	
TSMM15	016250	3331#																	
TSMM16	016610	3438#																	
TSMM6A	005520	1574#																	
TSMM6B	006262	1712#																	
TSMM6C	007152	1885#																	
TSMM6D	007762	2035#																	
TSM16A	016714	3461#	3502																
TSM16B	016746	3469#	3512																
TSM16C	016776	3477#	3522																
TSM16D	017042	3485#	3560																
TSM7	011176	2222	2234	2286#															
TSM9	011662	2385	2395	2413#	2463														
TSTLOC	001120	639#																	
TST1	002100	834#																	
TST10	006262	1716#																	
TST11	007152	1889#																	
TST12	007762	2039#																	
TST13	010572	2189#																	
TST14	011256	2312#																	
TST15	011522	2372#																	
TST16	012450	2596#																	
TST17	013064	2692#																	
TST2	002174	859#																	
TST20	013336	2749#																	
TST21	013742	2838#																	
TST22	014366	2968#																	
TST23	015064	3082#																	
TST24	016106	3298#																	
TST25	016250	3335#																	
TST26	016610	3442#																	
TST3	003550	1169#																	
TST4	004114	1262#																	
TST5	005210	1500#																	
TST6	005366	1544#																	
TST7	005520	1578#																	
TS10	012762	2612	2625	2636	2650	2658#													
TS11	013206	2700	2705	2710	2716#														
TS12	013660	2767	2774	2781	2788	2795	2802	2809#											
TS14	015500	3134	3156	3167#															
TS15	016420	3342	3350	3359	3384#														
TS16	017564	3610	3620#																
TS16A	017552	3614#	3638																
TS182P	014366	2960#																	
TS7	011046	2202	2209	2250#															
TS7FIN	011256	2246	2307#																
TS9FIN	012450	2409	2591#																
TYPDS =	104405	3692	4171#																
TYPE =	104401	809	3680	3690	3693	3762	3884	3953	3991	3992	3995	4008	4019	4038					

N7

GLOBAL AREAS MACY11 30A(1052) 20-MAR-84 11:31 PAGE 92
 KDJ11A.MAC 20-MAR-84 11:19 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0091

	4087	4090	4094	4167#	4252	4255	4258
TYPOC = 104402	3994	4168#	4257	4260			
TYPON = 104404	4170#						
TYPOS = 104403	4169#						
T10FIN 013064	2653	2687#					
T11FIN 013336	2712	2744#					
T12FIN 013742	2805	2831#					
T13FIN 014366	2889	2959#					
T14 015524	3152	3172#					
T14FIN 016106	2974	3161	3199	3289#			
T15 016474	3345	3353	3362	3403#			
T15A 016556	3367	3371	3375	3424#			
T15FIN 016610	3379	3436#					
UDPAR0 = 177660	371#						
UDPAR1 = 177662	372#						
UDPAR2 = 177664	373#						
UDPAR3 = 177666	374#						
UDPAR4 = 177670	375#						
UDPAR5 = 177672	376#						
UDPAR6 = 177674	377#						
UDPAR7 = 177676	378#						
UDPDR0 = 177620	349#	1594*	1904*				
UDPDR1 = 177622	350#						
UDPDR2 = 177624	351#						
UDPDR3 = 177626	352#						
UDPDR4 = 177630	353#						
UDPDR5 = 177632	354#						
UDPDR6 = 177634	355#						
UDPDR7 = 177636	356#						
UIPAR0 = 177640	360#						
UIPAR1 = 177642	361#						
UIPAR2 = 177644	362#						
UIPAR3 = 177646	363#						
UIPAR4 = 177650	364#						
UIPAR5 = 177652	365#						
UIPAR6 = 177654	366#						
UIPAR7 = 177656	367#						
UIPDR0 = 177600	338#	1730*	2054*				
UIPDR1 = 177602	339#						
UIPDR2 = 177604	340#						
UIPDR3 = 177606	341#						
UIPDR4 = 177610	342#						
UIPDR5 = 177612	343#						
UIPDR6 = 177614	344#						
UIPDR7 = 177616	345#						
VIR1 014250	2851	2919#					
VIR2 015736	3131	3234#					
VIR3 016054	3109	3133	3273#				
XBUF = 177566	491#						
XCSR = 177564	490#						
\$APTHD 000204	547	553#					
\$ASTAT = ***** U	4212	4227					
\$ATYC 022062	4183	4185#					
\$ATY1 022036	4181#						
\$ATY? 022044	3747	4182#					
\$ATY4 022054	4184#	4263					

PUSH	3230	3838	4185	4187	4208										
REPORT	3230														
SCOPE	2180														
SETPRI	3230														
SETTRA	41340	4159	4168	4169	4170	4171	4173	4175	4176	4177					
SETUP	1880	3230	767												
SKIP	3230														
SLASH	3230														
SPACE	3230														
STARS	3230	530	540	542	549	575	831	833	856	858	1166	1168	1259	1261	1497
	1499	1541	1543	1575	1577	1713	1715	1886	1888	2036	2038	2186	2188	2309	2311
	2369	2371	2593	2595	2689	2691	2746	2748	2833	2835	2837	2961	2963	2967	3077
	3079	3081	3290	3292	3297	3332	3334	3439	3441	3671	3721	3828	3895	3972	3975
	4043	4072	4113	4180	4235										
SWRSU	3230	7770													
TAIL	10	3668													
TRMTRP	41560														
TYPBIN	3230														
TYPDEC	3230	3691													
TYPNAM	3230														
TYPNUM	3230														
TYPOCS	3230														
TYP OCT	3230	3993													
TYP TXT	3230														
\$\$\$ESCA	3230														
\$\$\$NEWT	3230	831	850	1166	1259	1497	1541	1575	1713	1886	2036	2186	2309	2369	2593
	2689	2746	2833	2961	3077	3290	3332	3439							
\$\$\$SET	41400	4159	4168	4169	4170	4171	4173	4175	4176	4177					
\$\$\$SEIM	7930	800													
\$\$\$SKIP	3230														
.\$EQUAT	1880	213													
.\$HEADE	1900	195													
.\$KT11	1880	323													
.\$SETUP	1900	500													
.\$ACT1	1900	528													
.\$APT8	1880	573													
.\$APTH	1900	538													
.\$APTY	1910	4178													
.\$EOP	1880	3669													
.\$ERRO	1910														
.\$READ	1910	3970													
.\$TRAP	1900	4111													
.\$TYPD	1890	3826													
.\$TYPE	1890	3719													
.\$TYPO	1910	3893													
.\$4OCA	1880	501													

. ABS. 022472 000

ERRORS DETECTED: 0

CZKDKB/EN: ABS, CZKDKB, SEQ/CRF/DOC/SOL/NL; TOC=SYSMAC, SML/ML, CZKDKB, MAC/ML, KDJ11A, MAC
 RUN-TIME: 297 115 5 SECONDS
 RUN-TIME RATIO: 941/418=2.2

F8

GLOBAL AREAS MAC11 30A(1052) 20-MAR-84 11:31 PAGE 98
KDJ11A.MAC 20-MAR-84 11:19 CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0096

CORE USED: 52K (103 PAGES)

DOCUMENT PAGES: 96

.

KCJ11A.MAC	20-MAR....B1	KDJ11A.MAC	20-MAR....B5
KDJ11A.MAC	20-MAR....C1	KDJ11A.MAC	20-MAR....C5
KDJ11A.MAC	20-MAR....D1	KDJ11A.MAC	20-MAR....D5
KDJ11A.MAC	20-MAR....E1	KDJ11A.MAC	20-MAR....E5
KDJ11A.MAC	20-MAR....F1	KDJ11A.MAC	20-MAR....F5
KDJ11A.MAC	20-MAR....G1	KDJ11A.MAC	20-MAR....G5
KDJ11A.MAC	20-MAR....H1	KDJ11A.MAC	20-MAR....H5
KDJ11A.MAC	20-MAR....I1	KDJ11A.MAC	20-MAR....I5
KDJ11A.MAC	20-MAR....J1	KDJ11A.MAC	20-MAR....J5
KDJ11A.MAC	20-MAR....K1	KDJ11A.MAC	20-MAR....K5
KDJ11A.MAC	20-MAR....L1	KDJ11A.MAC	20-MAR....L5
KDJ11A.MAC	20-MAR....M1	KDJ11A.MAC	20-MAR....M5
KDJ11A.MAC	20-MAR....N1	KDJ11A.MAC	20-MAR....N5

KDJ11A.MAC	20-MAR....B2	KDJ11A.MAC	20-MAR....B6
KDJ11A.MAC	20-MAR....C2	KDJ11A.MAC	20-MAR....C6
KDJ11A.MAC	20-MAR....D2	KDJ11A.MAC	20-MAR....D6
KDJ11A.MAC	20-MAR....E2	KDJ11A.MAC	20-MAR....E6
KDJ11A.MAC	20-MAR....F2	KDJ11A.MAC	20-MAR....F6
KDJ11A.MAC	20-MAR....G2	KDJ11A.MAC	20-MAR....G6
KDJ11A.MAC	20-MAR....H2	KDJ11A.MAC	20-MAR....H6
KDJ11A.MAC	20-MAR....I2	KDJ11A.MAC	20-MAR....I6
KDJ11A.MAC	20-MAR....J2	KDJ11A.MAC	20-MAR....J6
KDJ11A.MAC	20-MAR....K2	KDJ11A.MAC	20-MAR....K6
KDJ11A.MAC	20-MAR....L2	KDJ11A.MAC	20-MAR....L6
KDJ11A.MAC	20-MAR....M2	KDJ11A.MAC	20-MAR....M6
KDJ11A.MAC	20-MAR....N2	KDJ11A.MAC	20-MAR....N6

KDJ11A.MAC	20-MAR....B3	KDJ11A.MAC	20-MAR....B7
KDJ11A.MAC	20-MAR....C3	KDJ11A.MAC	20-MAR....C7
KDJ11A.MAC	20-MAR....D3	KDJ11A.MAC	20-MAR....D7
KDJ11A.MAC	20-MAR....E3	KDJ11A.MAC	20-MAR....E7
KDJ11A.MAC	20-MAR....F3	KDJ11A.MAC	20-MAR....F7
KDJ11A.MAC	20-MAR....G3	KDJ11A.MAC	20-MAR....G7
KDJ11A.MAC	20-MAR....H3	KDJ11A.MAC	20-MAR....H7
KDJ11A.MAC	20-MAR....I3	KDJ11A.MAC	20-MAR....I7
KDJ11A.MAC	20-MAR....J3	KDJ11A.MAC	20-MAR....J7
KDJ11A.MAC	20-MAR....K3	KDJ11A.MAC	20-MAR....K7
KDJ11A.MAC	20-MAR....L3	KDJ11A.MAC	20-MAR....L7
KDJ11A.MAC	20-MAR....M3	KDJ11A.MAC	20-MAR....M7
KDJ11A.MAC	20-MAR....N3	KDJ11A.MAC	20-MAR....N7

KDJ11A.MAC	20-MAR....B4	KDJ11A.MAC	20-MAR....B8
KDJ11A.MAC	20-MAR....C4	KDJ11A.MAC	20-MAR....C8
KDJ11A.MAC	20-MAR....D4	KDJ11A.MAC	20-MAR....D8
KDJ11A.MAC	20-MAR....E4	KDJ11A.MAC	20-MAR....E8
KDJ11A.MAC	20-MAR....F4	KDJ11A.MAC	20-MAR....F8
KDJ11A.MAC	20-MAR....G4		
KDJ11A.MAC	20-MAR....H4		
KDJ11A.MAC	20-MAR....I4		
KDJ11A.MAC	20-MAR....J4		
KDJ11A.MAC	20-MAR....K4		
KDJ11A.MAC	20-MAR....L4		
KDJ11A.MAC	20-MAR....M4		
KDJ11A.MAC	20-MAR....N4		