

KD11-Z

11/44 TRAPS
CKKABBO

AH-F623B-MC
FICHE 1 OF 1

FEB 1981
COPYRIGHT © 79-80
MADE IN USA



The main body of the document is a microfiche card containing a grid of approximately 10 columns and 15 rows of data. Each cell in the grid contains a small, high-contrast image, likely a scan of a document page or a specific data entry. The text within these cells is extremely small and difficult to read, but the overall layout suggests a structured data table or a series of related documents.

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

.REM %

IDENTIFICATION

PRODUCT CODE: AC-F621B-MC
PRODUCT NAME: CKKABBO 11/44 TRAPS
DATE CREATED: OCTOBER 1980
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: DAN MILLEVILLE

COPYRIGHT (C) 1979,1980 DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

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

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.

122
123
124
125
126
127
128
129
130
131
132
133
134
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

1. ABSTRACT
THIS IS A TEST OF ALL OPERATIONS AND INSTRUCTIONS THAT CAUSE TRAPS. ALSO TESTED ARE TRAP OVERFLOW CONDITIONS, ODDITIES OF REGISTER 6, INTERRUPTS , THE RESET AND WAIT INSTRUCTIONS.
2. REQUIREMENTS
 - 2.1 EQUIPMENT
11/44 STANDARD COMPUTER
 - 2.2 STORAGE
 - 2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY FROM 0000 TO 17600.
3. LOADING PROCEDURE
 - 3.1 METHOD
PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.
4. STARTING PROCEDURE
THE PROGRAM STARTS AT 200.
IF IT IS DESIRED TO RESET THE PASS COUNT BACK TO ZERO ; THEN START THIS PROGRAM AT LOCATION 210
 - 4.2 PROGRAM AND/OR OPERATOR ACTION
LOAD PROGRAM INTO MEMORY. (BOTTOM 4K)
LOAD ADDRESS.
START.
THE PROGRAM WILL LOOP.
IT WILL PRINT "CKKABBO 11/44 TRAPS" AFTER THE FIRST ITERATION AND THEN PRINTS IT EVERY 15 TIMES (APROXIMATELY EVERY 15 SECONDS)

163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209

5. OPERATION

5.2 SUBROUTINE ABSTRACTS

5.2.1 BEGIN AT 200

5.2.2 SCOPE

IF A SCOPE LOOP IS NEEDED INSERT A BRANCH AS THE COMMENT TO THE HALT EXPLAINS.

5.2.3 TRAPCATCHER

THIS IS A SERIES OF INSTRUCTIONS DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS, THAT OCCUR IN THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.

THE PRINCIPLE OF THIS ROUTINE IS: THE VECTOR ENTRANCE ADDRESS POINTS TO THE NEXT SEQUENTIAL WORD WHICH WILL CONTAIN A HALT (000000) (THIS LOCATION IS ALSO THE STATUS WORD FOR THAT VECTOR ENTRANCE. BUT THIS WILL HAVE NO EFFECT ON IT ALSO BEING THE NEXT INSTRUCTION).

IF A HALT OCCURS IN THE TRAP OR INTERRUPT VECTOR AREA, REGISTER SIX SHOULD BE EXAMINED TO DETERMINE ITS CONTENTS, THEN USE REGISTER SIX CONTENTS AS AN ADDRESS TO DETERMINE WHERE THE PROGRAM WAS. WHEN THE INTERRUPT OR TRAP OCCURRED; MEMORY AS SPECIFIED BY R6 CONTAINS THE PC OF THE INSTRUCTION FOLLOWING THE INSTRUCTION WHERE THE TRAP OCCURRED.
THE CONTENTS OF LOCATION '\$TESTN'(304) CONTAINS THE TEST NUMBER THAT IT WAS DOING BEFORE IT TRAPPED.

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 LOADING AND STARTING AT 200 STARTS THE TEST. IF AN ERROR IS DETECTED, THERE WILL BE A HALT.
NOTE:IF A SCOPE LOOP IS NEEDED
THE COMMENT SECTION OF THE HALT EXPLAINS HOW TO UTILIZE THIS LOOP.

211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256

- 6. ERRORS
- 6.1 ALL ERRORS WILL CAUSE A HALT.
- 6.1.1 THE PROGRAM CHECKS TO SEE THAT THE P.C. DOESN'T JUMP
 WITHIN THE TESTS, BY A SEQUENCE COUNT CALLED '\$STSN'
 THIS TEST IS A SEQUENTIAL INCREMENT AND COMPARE COUNT.

EXAMPLE

```
TSTA:  INC   $STSNM      ;INCREMENT THE TEST NUMBER
        CMP   #A,$STSNM  ;COMPARE FOR THE RIGHT TEST
        BNE  TSTA+1-12   ;IF NOT CORRECT BRACHN TO A HALT
        ----
        CODE
```

IMPORTANT

IF AN ERROR IS DETECTED ;IT COULD BE BECAUSE OF TWO REASONS.
A) WRONG TEST NUMBER
B) ERROR IN THE PRESENT TEST.

////////////////////////////////////
THE TEST SEQUENCE LOCATION 'TESTN' SHOULD BE CHECKED FIRST
TO SEE IF IT MATCHES THE PRESENT TEST.
IF IT DOESN'T MATCH ; THEN THE CONTENTS OF THIS LOCATION
TELL YOU WHICH TEST IT WAS DOING BEFORE IT HALTED.
////////////////////////////////////

- 6.2 ERROR RECOVERY
- ON TRAP ERRORS - RESTART AT STARTING ADDRESS

- 7. RESTRICTIONS
- 7.1 STARTING RESTRICTION

NONE

- 7.2 OPERATIONAL RESTRICTION

NONE

258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313

8. MISCELLANEOUS

8.1 EXECUTION TIME

1ST PASS APPROX. 2 SEC., THEREAFTER EVERY 15 SEC

9. PROGRAM DESCRIPTION

THIS PROGRAM CHECKS THAT ON ALL TRAP OPERATIONS REGISTER 6 IS DECREMENTED THE CORRECT AMOUNT, THAT THE CORRECT PC IS SAVED ON THE STACK, THAT THE OLD CONDITION CODES AND PRIORITY ARE PLACED ON THE STACK AND THAT THE NEW STATUS AND CONDITION CODES ARE CORRECT. BOTH THE "TRAP" AND "EMT" TRAP INSTRUCTIONS ARE TESTED TO SEE THAT ALL COMBINATIONS WILL TRAP. CHECKED ALSO IS THAT ALL RESERVED INSTRUCTIONS WILL TRAP. VERIFICATION OF THE "TRT" INSTRUCTION (00003) WHICH IS USED FOR SOFTWARE DEBUG ROUTINES: ODT,DDT, IS DONE. ALSO, THE TRACE BIT IS CHECKED TO SEE IF IT CAUSES A TRAP. THE RTI AND RTT INSTRUCTIONS ARE CHECKED. STACK OVERFLOW IS ALSO CHECKED FOR ALL THE TRAP INSTRUCTIONS. SPECIAL CHECKS ARE MADE TO SEE IF BUS ERROR TRAPS OCCUR ON NON-EXISTENT MEMORY. PIRQ TRAPS ARE CHECKED AT ALL LEVELS

10.0 RUNNING UNDER APT

THE EXECUTION TIMES PROVIDED IN THE APT SCRIPT THAT FOLLOWS ARE FOR EXECUTION WITH A 11/44 PROCESSOR, CACHE, 16K CORE MEMORY, AND 300 BAUD.

THE FOLLOWING IS A PROGRAM LOAD FILE USED BY APT:

1. E TABLE 'A' IS USED FOR APT DUMP MODE.
2. E TABLE 'B' IS USED FOR APT QV AND RUN TIME MODES. \$ENVM=040 INDICATES THAT TYPEOUTS WILL BE SUPRESSED.

	1ST PASS RUN TIME	LONGEST TEST TIME	ADDITIONAL RUN TIME
	5 .	5	0
.....		E TABLES
E-MODE/S-MODE (\$ENVM/\$ENV)		A 000/000	B 040/001
SWITCH REGISTER 1 (\$SWREG)		000000	000000
SWITCH REGISTER 2 CPU TYPE/OPTIONS		000000 00/0000	000000 00/0000

314
315
316
317
318
319
320

11.0

REVISION HISTORY

REVISION	DATE	COMMENT
CKKABA	MARCH 1979	ORIGINAL RELEASE
CKKABB	NOVEMBER 1980	FIX DIAGNOSTIC HALT WITH CIS SWITCH IN MAINT POSITIO

x

```

321                   :ALL INSTRUCTIONS THAT ARE RESERVED
322                   :SHOULD TRAP TO LOCATION 10, AND THE
323                   :PC THAT POINTS TO THE TRAPPING INSTRUCTION
324                   :SHOULD BE PLACED ON THE STACK
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364

```

```

:LISTING

```

```

000000

```

```

000006
000006
000003
000001
000005
000002
000000
000003
000004
000014
000030
000020
000034
177564
177560
177564
177566
000240
000240
177776
000010
000010
004700
000100
177776
177766
177413
177776

```

```

.LIST ME
.NLIST MC,MD,CND
.ENABLE ABS
SP=%6
R6=%6
TAB=%3
LAST=%1
FIRST=%5
R2=%2
HLT=HALT
TRT=3
RTRAP5=4
RTRAP4=14
RTRAP3=30
RTRAP2=20
RTRAP1=34
TTCSR=177564
TRCSR=177560
TPS=177564
TPB=177566
BELL=240
NOP=240
STATUS=177776
TRAPA=10
RTRAP=10
ILLA=004700
ILLB=100
CC=177776
CPUERR=177766
CERMSK=177413
PSW=177776

```

```

:ILLEGAL ADDRESSES
:FOR TRACE TRAP
:FOR EMULATOR TRAP
:FOR IOT TRAP
:FOR TRAP INST

```

```

.MCALL .SAPTHDR
.MCALL .SAPTBL5
.MCALL .SACT11

```


436 000200
 437 000200 000137 000622
 438 000210
 439 000210 005037 000306
 440 000214 000137 000622
 441 000300
 442

```

.=200
JMP BEGIN
.=210
CLR $PASS
JMP BEGIN
.=300
.SBTTL ACT11 HOOKS
*****
;HOOKS REQUIRED BY ACT11
$SVPC=.;SAVE PC
.=46
$ENDAD ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
.=52
.WORD 0 ;;2)SET LOC.52 TO ZERO
.= $SVPC ;; RESTORE PC
    
```

443

000300
 000300 000000
 000302 000000
 000304 000000
 000306 000000
 000310 000000
 000312 000000
 000314 000000
 000316 000000
 000320
 000320 000
 000321 000
 000322 000000
 000324 000000
 000326 000000

```

.SBTTL APT MAILBOX-ETABLE
*****
.EVEN
$MAIL: ;;APT MAILBOX
$MSGTY: .WORD AMSTY ;;MESSAGE TYPE CODE
$FATAL: .WORD AFATAL ;;FATAL ERROR NUMBER
$TESTN: .WORD ATESTN ;;TEST NUMBER
$PASS: .WORD APASS ;;PASS COUNT
$DEVCT: .WORD ADEVCT ;;DEVICE COUNT
$UNIT: .WORD AUNIT ;;I/O UNIT NUMBER
$MSGAD: .WORD AMSGAD ;;MESSAGE ADDRESS
$MSGLG: .WORD AMSGLG ;;MESSAGE LENGTH
$ETABLE: ;;APT ENVIRONMENT TABLE
$ENV: .BYTE AENV ;;ENVIRONMENT BYTE
$ENVM: .BYTE AENVM ;;ENVIRONMENT MODE BITS
$SWREG: .WORD ASWREG ;;APT SWITCH REGISTER
$USWR: .WORD AUSWR ;;USER SWITCHES
$CPUOP: .WORD ACPUOP ;;CPU TYPE,OPTIONS
BITS 15-11=CPU TYPE
11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
11/70=06,PDQ=07,Q=10
BIT 10=REAL TIME CLOCK
BIT 9=FLOATING POINT PROCESSOR
BIT 8=MEMORY MANAGEMENT
    
```

000330

444

000330
 000024 000024
 000024 000200
 000044 000044
 000044 000330
 000330
 000330 000000
 000332 000300

```

$ETEND:
.MEXIT
.SBTTL APT PARAMETER BLOCK
*****
;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
*****
.$X=.;SAVE CURRENT LOCATION
.=24 ;;SET POWER FAIL TO POINT TO START OF PROGRAM
200 ;;FOR APT START UP
.=44 ;;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR ;;POINT TO APT HEADER BLOCK
.=.$X ;;PESET LOCATION COUNTER
*****
;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
;INTERFACE SPEC.
$APTHD:
$HIBTS: .WORD 0 ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBADR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
    
```

000334	000005	\$TSTM: .WORD	5	::RUN TIM OF LONGEST TEST
000336	000005	\$PASTM: .WORD	5	::RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
000340	000000	\$UNITM: .WORD	0	::ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
000342	000014	.WORD		\$ETEND-\$MAIL/2 ::LENGTH MAILBOX-ETABLE(WORDS)
445	000304	\$TSTNM=\$TESTN		
446	000302	\$ERROR=\$FATAL		
447				
448	000500		=500	
449	000500	BUFF: 0		
450	000502	RCPUER: .WORD	0	
451	000504	SR0:	177572	
452	000506	SR0H:	177573	
453	000510	SR1:	177574	
454	000512	SR2:	177576	
455	000514	KTVEC:	250	
456	000516	KTSTA:	252	
457	000520	ADRTAB:		
458	000520	UPDR0:	177600	;USER PAGE DESCRIPTOR REGISTERS
459	000522	UPDR1:	177602	
460	000524	UPDR2:	177604	
461	000526	UPDR3:	177606	
462	000530	UPDR4:	177610	
463	000532	UPDR5:	177612	
464	000534	UPDR6:	177614	
465	000536	UPDR7:	177616	
466		:		
467	000540	UPAR0:	177640	;USER PAGE ADDRESS REGISTERS
468	000542	UPAR1:	177642	
469	000544	UPAR2:	177644	
470	000546	UPAR3:	177646	
471	000550	UPAR4:	177650	
472	000552	UPAR5:	177652	
473	000554	UPAR6:	177654	
474	000556	UPAR7:	177656	
475		:		
476	000560	KPDR0:	172300	;KERNEL PAGE DESCRIPTOR REGISTERS
477	000562	KPDR1:	172302	
478	000564	KPDR2:	172304	
479	000566	KPDR3:	172306	
480	000570	KPDR4:	172310	
481	000572	KPDR5:	172312	
482	000574	KPDR6:	172314	
483	000576	KPDR7:	172316	
484		:		
485	000600	KPAR0:	172340	;KERNEL PAGE ADDRESS REGISTERS
486	000602	KPAR1:	172342	
487	000604	KPAR2:	172344	
488	000606	KPAR3:	172346	
489	000610	KPAR4:	172350	
490	000612	KPAR5:	172352	
491	000614	KPAR6:	172354	
492	000616	KPAR7:	172356	
493	000620	ADREND:	.-2	
494				
495				
496				

```

498
499
500 000622 012706 000500 BEGIN: MOV #500,%6 ;SET UP SACK BUFF
501 000626 012737 177777 017356 MOV #-1,PASSPT ;CLEAR THE ITERATION COUNTER
502 000634 023737 000042 017330 CMP 42,$ENDAD
503 000642 001404 BEQ RESTRT
504 000644 012700 017566 MOV #TITLE,R0
505 000650 004737 020000 JSR PC,PRTMSG
506 000654 005037 000300 RESTRT: CLR $MSGTY
507 000660 012706 000500 MOV #500,%6
508 000664 012737 017712 000024 MOV #PWRDWN,24 ;SET UP THE POWER DOWN VECTOR
509 000672 012737 000340 000026 MOV #340,26 ;SET UP POWER DOWN PRIORITY
510 000700 012737 000006 000004 MOV #6,4 ;SET UP TRAP VECTORS 4 & 6.
511 000706 005037 000006 CLR 6
512 000712 012737 000012 000010 MOV #12,10
513 000720 005037 000012 CLR 12
514 000724 005037 000304 CLR $TSTNM
515 000730 005037 000302 CLR $ERROR
516 000734 012702 000300 MOV # $MSGTY,R2
517
518 ;SPECIAL CASE OF ODD;.EVEN .BYTE AND REGISTER 6
519 000000 HERE=0
520
521 000740 000412 BR TST1
522 000742 000000 K1: 0
523 000744 000000 K2: 0
524 000746 000000 K3: 0
525 000750 000000 K4: 0
526 000752 000000 K5: 0
527 000754 000000 K6: 0
528 000756 052525 K7: 052525
529 000760 052400 K10: 052400
530 000762 000000 K11: 0
531 000764 000000 K12: 0

```



```

;AND REPLACE NEXT INST WITH 710
559
560 001160 005006          BR5:  CLR    %6
561 001162 005004          CLR    %4
562 001164 122624          CMPB   (6)+,(4)+      ;TEST INCREMENT OF R4
563 001166 020427 000001  CMP    %4,#1
564 001172 001405          BEQ    BR6
      001174 012737 000006 000302  MOV    #6,$FATAL
      001202 005212          INC    (R2)
      001204 000000          HALT
;MOVE TO MAILBOX # ***** 6 *****
;SET MSGTYP TO FATAL ERROR
;WRONG INCREMENT OF R4
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 675

565 001206 005006          BR6:  CLR    %6
566 001210 005004          CLR    %4
567 001212 122426          CMPB   (4)+,(6)+      ;TEST INCREMENT OF R6
568 001214 020627 000002  CMP    %6,#2
569 001220 001405          BEQ    BR7
      001222 012737 000007 000302  MOV    #7,$FATAL
      001230 005212          INC    (R2)
      001232 000000          HALT
;MOVE TO MAILBOX # ***** 7 *****
;SET MSGTYP TO FATAL ERROR
;WRONG INCREMENT OF R6
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 662

570
571 001234 005006          BR7:  CLR    %6
572 001236 005004          CLR    %4
573 001240 122426          CMPB   (4)+,(6)+      ;TEST INCREMENT OF R4
574 001242 020427 000001  CMP    %4,#1
575 001246 001405          BEQ    BR10
      001250 012737 000010 000302  MOV    #10,$FATAL
      001256 005212          INC    (R2)
      001260 000000          HALT
;MOVE TO MAILBOX # ***** 10 *****
;SET MSGTYP TO FATAL ERROR
;WRONG INCREMENT OF R4
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 647

576
577 001262 012706 001000          BR10: MOV    #1000,%6
578 001266 124627 000000          CMPB   -(6),#HERE
579 001272 022706 000776          CMP    #776,%6
580 001276 001405          BEQ    TST2
      001300 012737 000011 000302  MOV    #11,$FATAL
      001306 005212          INC    (R2)
      001310 000000          HALT
;TEST DECREMENT OF R6
;MOVE TO MAILBOX # ***** 11 *****
;SET MSGTYP TO FATAL ERROR
;WRONG DECREMENT OF R6,OR WRONG $TSTNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 633

```

.SBTTL TEST #2 - TEST TRANSFER OF .BYTE USING R6
:*****
:TEST 2 - TEST TRANSFER OF .BYTE USING R6
:*****

001312 005237 000304
001316 022737 000002 000304
001324 001137
582 001326 012737 123456 000752
583 001334 012737 050505 000742
584 001342 012705 000742
585 001346 012706 000752
586 001352 112625
587 001354 022737 050456 000742
588 001362 001405
001364 012737 000012 000302
001372 005212
001374 000000

TST2: INC \$TESTN ;UPDATE TEST NUMBER
CMP #2,\$TESTN ;SEQUENCE ERROR?
BNE TST3-12 ;BR TO ERROR HALT ON SEQ ERROR
MOV #123456,K5
MOV #050505,K1
MOV #K1,%5 ;%5=(050505)K1
MOV #K5,%6 ;%6=(123456)K5
MOVB (6)+,(5)+ ;LOW .BYTE OF R6 TO R5
CMP #050456,K1
BEQ BR11
MOV #12,\$FATAL ;MOVE TO MAILBOX # ***** 12 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;FALSE TRANSFER OF .BYTE
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 753

589
590 001376 012737 123456 000752
591 001404 012737 050505 000742
592 001412 012705 000742
593 001416 012706 000754
594 001422 114625
595 001424 023727 000742 050456
596 001432 001405
001434 012737 000013 000302
001442 005212
001444 000000

BR11: MOV #123456,K5
MOV #050505,K1
MOV #K1,%5 ;%5(050505)K1
MOV #K6,%6 ;%6(123456)K5
MOVB -(6),(5)+ ;LOW .BYTE OF R6 TO R5 (DECREMENT)
CMP K1,#050456
BEQ BR12
MOV #13,\$FATAL ;MOVE TO MAILBOX # ***** 13 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;FALSE R6 .BYTE TRANSFER
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 727

597
598 001446 012737 123456 000742
599 001454 012737 050505 000752
600 001462 012705 000742
601 001466 012706 000752
602 001472 112526
603 001474 022737 050456 000752
604 001502 001405
001504 012737 000014 000302
001512 005212
001514 000000

BR12: MOV #123456,K1
MOV #050505,K5
MOV #K1,%5 ;(123456)
MOV #K5,%6 ;(050505)
MOVB (5)+,(6)+ ;LOW OF R5 TO LOW OF R6
CMP #050456,K5
BEQ BR13
MOV #14,\$FATAL ;MOVE TO MAILBOX # ***** 14 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;FALSE R6 .BYTE TRANSFER
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 703

605
606 001516 012737 123456 000742
607 001524 012737 050505 000752
608 001532 012705 000743
609 001536 012706 000752
610 001542 112526
611 001544 023727 000752 050647
612 001552 001405
001554 012737 000015 000302
001562 005212
001564 000000

BR13: MOV #123456,K1
MOV #050505,K5
MOV #K1+1,%5 ;123456
MOV #K5,%6 ;050505
MOVB (5)+,(6)+ ;HIGH OF R5 TO LOW OF R6
CMP K5,#050647
BEQ BR14
MOV #15,\$FATAL ;MOVE TO MAILBOX # ***** 15 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;FALSE R6 .BYTE TRANSFER
;TO SCOPE REPLACE HALT WITH 240

:AND REPLACE NEXT INST WITH 657

```

613
614 001566 012737 123456 000742 BR14: MOV #123456,K1
615 001574 012737 050505 000752 MOV #050505,K5
616 001602 012705 000743 MOV #K1+1,%5
617 001606 012706 000752 MOV #K5,%6
618 001612 112625 MOVB (6)+,(5)+
619 001614 022737 042456 000742 CMP #042456,K1
620 001622 001405 BEQ TST3
    001624 012737 000016 000302 MOV #16,$FATAL
    001632 005212 INC (R2)
    001634 000000 HALT

```

:R5-123456-ODD ADDRESS
:R6-050505--.EVEN ADDRESS
:LOW OF R6 TO HIGH OF R5

:MOVE TO MAILBOX # ***** 16 *****
:SET MSGTYP TO FATAL ERROR
:FAILED LOW OF 6 TO HIGH OF 5,OR WRONG \$STNM
:TO SCOPE REPLACE HALT WITH 240
:AND REPLACE NEXT INST WITH 633

TEST #3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS

.SBTTL TEST #3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
:*****
:TEST 3 - TEST BYTE OPERATION WITH SEQ ODD-EVEN ADDRESS
:*****

```

621 001636 005237 000304
001642 022737 000003 000304
001650 001103
622 001652 123737 000756 000757
623 001660 001405
001662 012737 000017 000302
001670 005212
001672 000000

TST3: INC $TESTN ;UPDATE TEST NUMBER
CMP #3,$TESTN ;SEQUENCE ERROR?
BNE TST4-12 ;BR TO ERROR HALT ON SEQ ERROR
CMPB K7,K7+1 ;SAME .WORD LOW TO HIGH
BEQ BR15
MOV #17,$FATAL ;MOVE TO MAILBOX # ***** 17 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;SHOULD COMPARE LOW TO HIGH
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 766

624
625 001674 123737 000757 000756 BR15: CMPB K7+1,K7 ;COMPARE ODD TO .EVEN SAME .WORD
626 001702 001405 BEQ BR16
001704 012737 000020 000302 MOV #20,$FATAL ;MOVE TO MAILBOX # ***** 20 *****
001712 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
001714 000000 HALT ;ODD TO .EVEN .BYTE FAILURE
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 755

627
628 001716 123737 000761 000756 BR16: CMPB K10+1,K7 ;SEQUENTIAL .BYTES
629 001724 001405 BEQ BR17
001726 012737 000021 000302 MOV #21,$FATAL ;MOVE TO MAILBOX # ***** 21 *****
001734 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
001736 000000 HALT ;ODD TO .EVEN FAILED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 744

630
631 001740 123737 000760 000754 BR17: CMPB K10,K6
632 001746 001405 BEQ BR20
001750 012737 000022 000302 MOV #22,$FATAL ;MOVE TO MAILBOX # ***** 22 *****
001756 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
001760 000000 HALT ;.EVEN TO EVEN FAILED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 733

633 001762 123737 000757 000761 BR20: CMPB K7+1,K10+1
634 001770 001405 BEQ BR21
001772 012737 000023 000302 MOV #23,$FATAL ;MOVE TO MAILBOX # ***** 23 *****
002000 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002002 000000 HALT ;ODD TO ODD FAILED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 722

635
636 002004 123737 000760 000761 BR21: CMPB K10,K10+1
637 002012 001005 BNE BR22
002014 012737 000024 000302 MOV #24,$FATAL ;MOVE TO MAILBOX # ***** 24 *****
002022 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002024 000000 HALT ;LOW TO HIGH IN SAME .WORD FAILED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 711

638
639 002026 123737 000761 000761 BR22: CMPB K10+1,K10+1
640 002034 001405 BEQ BR23
002036 012737 000025 000302 MOV #25,$FATAL ;MOVE TO MAILBOX # ***** 25 *****

```

002044 005212
002046 000000

INC (R2)
HALT

;SET MSGTYP TO FATAL ERROR
;HIGH TO LOW IN SAME WORD FAILED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 700

641

642 002050 123737 000760 000757 BR23:
643 002056 001005
002060 012737 000026 000302
002066 005212
002070 000000

CMPB K10,K7+1
BNE TST4
MOV #26,\$FATAL
INC (R2)
HALT

;MOVE TO MAILBOX # ***** 26 *****
;SET MSGTYP TO FATAL ERROR
;. EVEN TO ODD FAILED,OR WRONG \$TSTNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 667

.SBTTL TEST #4 - TEST THE CC BITS
:*****
:TEST 4 - TEST THE CC BITS
:*****

```

002072 005237 000304 000304 TST4: INC $TESTN ;UPDATE TEST NUMBER
002076 022737 000004 000304 CMP #4,$TESTN ;SEQUENCE ERROR?
645 002104 001062 BNE TST5-12 ;BR TO ERROR HALT ON SEQ ERROR
646 002106 000277 SCC ;SET STATUS
646 002110 005037 177776 CLR STATUS ;CLEAR STATUS
647 002114 103005 BCC BR33
002116 012737 000027 000302 MOV #27,$FATAL ;MOVE TO MAILBOX # ***** 27 *****
002124 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002126 000000 HALT ;C NOT CLEAR
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 766

648 002130 BR33: BVC BR34
002130 102005 BVC BR34
002132 012737 000030 000302 MOV #30,$FATAL ;MOVE TO MAILBOX # ***** 30 *****
002140 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002142 000000 HALT ;V NOT CLEAR
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 760

649 002144 BR34: BNE BR35
002144 001005 BNE BR35
002146 012737 000031 000302 MOV #31,$FATAL ;MOVE TO MAILBOX # ***** 31 *****
002154 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002156 000000 HALT ;Z NOT CLEAR
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 752

650 002160 BR35: BPL BR36
002160 100005 BPL BR36
002162 012737 000032 000302 MOV #32,$FATAL ;MOVE TO MAILBOX # ***** 32 *****
002170 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002172 000000 HALT ;N NOT CLEAR
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 744

651 002174 000257 BR36: CCC
652 002176 052737 000017 177776 BIS #17,STATUS ;CLEAR CONDITION CODES
653 ;SET STATUS TO ONES
654 002204 103405 BCS BR37
002206 012737 000033 000302 MOV #33,$FATAL ;MOVE TO MAILBOX # ***** 33 *****
002214 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002216 000000 HALT ;C NOT SET
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 732

655 002220 BR37: BVS BR40
002220 102405 BVS BR40
002222 012737 000034 000302 MOV #34,$FATAL ;MOVE TO MAILBOX # ***** 34 *****
002230 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002232 000000 HALT ;V NOT SET
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 724

656 002234 BR40: BEQ BR41
002234 001405 BEQ BR41
002236 012737 000035 000302 MOV #35,$FATAL ;MOVE TO MAILBOX # ***** 35 *****
002244 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
002246 000000 HALT ;Z NOT SET

```

```

;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 716

657 002250          BR41:
    002250 100405
    002252 012737 000036 000302
    002260 005212
    002262 000000
        BMI      TST5
        MOV      #36,$FATAL
        INC      (R2)
        HALT
;MOVE TO MAILBOX # ***** 36 *****
;SET MSGTYP TO FATAL ERROR
;N NOT SET,OR WRONG $STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 710

```

658

.SBTTL TEST #5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
:*****
:TEST 5 - TEST THAT A TRAP OCCURS ON A RESERVED INS
:*****

002264 005237 000304
002270 022737 000005 000304
002276 001006
659 002300 012706 000500
660 002304 012737 002326 000010
661 002312 000010
662 002314
002314 012737 000037 000302
002322 005212
002324 000000

TST5: INC \$TESTN ;UPDATE TEST NUMBER
CMP #5,\$TESTN ;SEQUENCE ERROR?
BNE RETA ;BR TO ERROR HALT ON SEQ ERROR
MOV #BUFF,SP ;STACK POINTER SETUP
MOV #RETAH,RTRAP ;RETURN LOCATION
TRAPA ;RESERVED INSTRUCTION, SHOULD TRAP

RETA: MOV #37,\$FATAL ;MOVE TO MAILBOX # ***** 37 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;RESERVE INSTRUCTION DIDN'T TRAP,OR WRONG \$STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 764

RETAH:

663 002326

664

.SBTTL TEST #6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****

002326 005237 000304
002332 022737 000006 000304
002340 001011
665 002342 012706 000500
666 002346 012737 002356 000010
667 002354 000010
668 002356 020627 000474
669 002362 001405
002364 012737 000040 000302
002372 005212
002374 000000

TST6: INC \$TESTN ;UPDATE TEST NUMBER
CMP #6,\$TESTN ;SEQUENCE ERROR?
BNE TST7-12 ;BR TO ERROR HALT ON SEQ ERROR
MOV #BUFF,SP ;STACK POINTER SETUP
MOV #RETB,RTRAP ;RETURN POINTER
TRAPA ;RESERVED INSTRUCTION
RETB: CMP SP,#BUFF-4 ;TEST DECREMENT OF SP
BEQ TST7
MOV #40,\$FATAL ;MOVE TO MAILBOX # ***** 40 *****
INC (R2) ;SET MSGTYP TO FATAL ERROR
HALT ;NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 761

670

```

.SBTTL TEST #7 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 7 - TEST THAT PROPER P.C. IS SAVED
:*****
TST7:  INC      $TESTN      ;UPDATE TEST NUMBER
      CMP      #7,$TESTN   ;SEQUENCE ERROR?
      BNE     TST10-12    ;BR TO ERROR HALT ON SEQ ERROR
      MOV     #BUFF,SP    ;STACK POINTER SETUP
      MOV     #RETC,RTRAP ;RETURN FROM TRAP POINTER
INSTC: TRAPA              ;TRAP ON THIS INSTRUCTION
RETC:  CMP     #.,BUFF-4   ;CHECK FOR INCREMENTED P.C.
      BEQ     TST10
      MOV     #41,$FATAL  ;MOVE TO MAILBOX # ***** 41 *****
      INC     (R2)        ;SET MSGTYP TO FATAL ERROR
      HALT              ;INCORRECT P.C.,OR WRONG $STNM
                          ;TO SCOPE REPLACE HALT WITH 240
                          ;AND REPLACE NEXT INST WITH 760

```

```

002376 005237 000304
002402 022737 000007 000304
002410 001012
671 002412 012706 000500
672 002416 012737 002426 000010
673 002424 000010
674 002426 022737 002426 000474
675 002434 001405
    002436 012737 000041 000302
    002444 005212
    002446 000000

```

676

```
.SBTTL TEST #10 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
:TEST 10 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
TST10:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #10,$TESTN  ;SEQUENCE ERROR?
        BNE     TST11-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;SET UP
        MOV     #RETD,RTRAP ;SET UP
        CLR     CC          ;CLEAR CC AND PRIORITY
        TRAPA                    ;TRAP
RETD:   CMP     BUFF-2,#0    ;TEST THAT OLD STATUS WENT TO STACK
        BEQ     1$
        MOV     #42,$FATAL  ;MOVE TO MAILBOX # ***** 42 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                    ;INCORRECT STATUS
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 755
1$:     MOV     #BUFF,SP    ;SET UP
        MOV     #RETE,RTRAP ;SET UP
        MOV     #357,CC     ;SET PRIORITY
        TRAPA                    ;SET CC
                                ;TRAP
RETE:   CMP     BUFF-2,#357 ;COMPARES STATUS ON STACK
        BEQ     TST11
        MOV     #43,$FATAL  ;MOVE TO MAILBOX # ***** 43 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                    ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 732
```

```
002450 005237 000304
002454 022737 000010 000304
002462 001040
677 002464 012706 000500
678 002470 012737 002506 000010
679 002476 005037 177776
680 002502 000257
681 002504 000010
682 002506 023727 000476 000000
683 002514 001405
    002516 012737 000042 000302
    002524 005212
    002526 000000

684 002530 012706 000500
685 002534 012737 002554 000010
686 002542 012737 000357 177776
687 002550 000277
688 002552 000010
689 002554 023727 000476 000357
690 002562 001405
    002564 012737 000043 000302
    002572 005212
    002574 000000
```



```

709 002772          1$:
    002772 001405
    002774 012737 000052 000302
    003002 005212
    003004 000000
        BEQ      2$
        MOV      #52,$FATAL
        INC      (R2)
        HALT
        ;MOVE TO MAILBOX # ***** 52 *****
        ;SET MSGTYP TO FATAL ERROR
        ;Z NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 701

710 003006          2$:
    003006 102405
    003010 012737 000053 000302
    003016 005212
    003020 000000
        BVS      3$
        MOV      #53,$FATAL
        INC      (R2)
        HALT
        ;MOVE TO MAILBOX # ***** 53 *****
        ;SET MSGTYP TO FATAL ERROR
        ;V NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 673

711 003022          3$:
    003022 103405
    003024 012737 000054 000302
    003032 005212
    003034 000000
        BCS      4$
        MOV      #54,$FATAL
        INC      (R2)
        HALT
        ;MOVE TO MAILBOX # ***** 54 *****
        ;SET MSGTYP TO FATAL ERROR
        ;C NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 665

712 003036 013706 177776
713 003042 042706 000017
714 003046 022706 000340
715 003052 001405
    003054
    003054 012737 000055 000302
    003062 005212
    003064 000000
        4$:
        MOV      CC,SP
        BIC      #17,SP
        CMP      #340,SP
        BEQ      STPPA
        STPP:
        MOV      #55,$FATAL
        INC      (R2)
        HALT
        ;MOVE TO MAILBOX # ***** 55 *****
        ;SET MSGTYP TO FATAL ERROR
        ;PRIORITY WAS CHANGED,OR WRONG $STNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 651

716 003066 012737 000012 000010 STPPA: MOV      #12,10
/17 003074 005037 000012          CLR      12
  
```

718

```

.SBTTL TEST #12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
:*****
:TEST 12 - TEST THAT A TRAP OCCURS FOR A "TRAP" INSTRUCTION
:*****
TST12:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #12,$TESTN ;SEQUENCE ERROR?
        BNE     TST13-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #12,10
        CLR     12
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETA1,RTRAP1 ;RETURN LOCATION
        TRAP
        MOV     #56,$FATAL  ;RESERVED INSTRUCTION, SHOULD TRAP
        INC     (R2)        ;MOVE TO MAILBOX # ***** 56 *****
        HALT               ;SET MSGTYP TO FATAL ERROR
                               ;TRAP DIDN'T TRAP,OR WRONG $STNM
                               ;TO SCOPE REPLACE HALT WITH 240
                               ;AND REPLACE NEXT INST WITH 757
  
```

```

003100 005237 000304
003104 022737 000012 000304
003112 001013
719 003114 012737 000012 000010
720 003122 005037 000012
721 003126 012706 000500
722 003132 012737 003154 000034
723 003140 104400
724 003142 012737 000056 000302
    003150 005212
    003152 000000
  
```

725 003154

RETA1:

726

```

.SBTTL TEST #13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
TST13:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #13,$TESTN  ;SEQUENCE ERROR?
        BNE     TST14-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETB1,RTRAP1 ;RETURN POINTER
        TRAP   ;RESERVED INSTRUCTION
RETB1:  CMP     SP,#BUFF-4   ;TEST DECREMENT OF SP
        BEQ     TST14
        MOV     #57,$FATAL  ;MOVE TO MAILBOX # ***** 57 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT   ;NOT DECREMENTED TWO WORDS,OR WRONG $STNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761
  
```

```

003154 005237 000304
003160 022737 000013 000304
003166 001011
727 003170 012706 000500
728 003174 012737 003204 000034
729 003202 104400
730 003204 020627 000474
731 003210 001405
    003212 012737 000057 000302
    003220 005212
    003222 000000
  
```

732

```
.SBTTL TEST #14 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 14 - TEST THAT PROPER P.C. IS SAVED
:*****
TST14: INC $TESTN ;UPDATE TEST NUMBER
      CMP #14,$TESTN ;SEQUENCE ERROR?
      BNE TST15-12 ;BR TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #RETC1,RTRAP1 ;RETURN FROM TRAP POINTER
      TRAP ;TRAP ON THIS INSTRUCTION
RETC1: CMP #,$BUFF-4 ;CHECK INCREMENTED P.C.
      BEQ TST15
      MOV #60,$FATAL ;MOVE TO MAILBOX # ***** 60 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT P.C.,OR WRONG $TESTNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 760
```

```
003224 005237 000304
003230 022737 000014 000304
003236 001012
733 003240 012706 000500
734 003244 012737 003254 000034
735 003252 104400
736 003254 022737 003254 000474
737 003262 001405
003264 012737 000060 000302
003272 005212
003274 000000
```


TEST #16 - TEST THAT 'NEW' STATUS IS CORRECT

752

.SBTTL TEST #16 - TEST THAT 'NEW' STATUS IS CORRECT
:*****
:TEST 16 - TEST THAT 'NEW' STATUS IS CORRECT
:*****

003422 005237 000304 000304 TST16: INC \$TESTN ;UPDATE TEST NUMBER
003426 022737 000016 000304 CMP #16,\$TESTN ;SEQUENCE ERROR?
003434 001121 BNE TST17-12 ;BR TO ERROR HALT ON SEQ ERROR
753 003436 012706 000500 MOV #BUFF,SP
754 003442 012737 003456 000034 MOV #RETG1,RTRAP1
755 003450 005037 000036 CLR RTRAP1+2 ;CLEAR FUTURE PRIORITY AND CC
756 003454 104400 TRAP
757 003456 RETG1: ;TEST FOR 'C' CLEARED
758 003456 100005 BPL 1\$
003460 012737 000063 000302 MOV #63,\$FATAL ;MOVE TO MAILBOX # ***** 63 *****
003466 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003470 000000 HALT ;C NOT CLEARED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 761
759 003472 1\$:
003472 001005 BNE 2\$
003474 012737 000064 000302 MOV #64,\$FATAL ;MOVE TO MAILBOX # ***** 64 *****
003502 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003504 000000 HALT ;Z NOT CLEARED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 753
760 003506 2\$:
003506 102005 BVC 3\$
003510 012737 000065 000302 MOV #65,\$FATAL ;MOVE TO MAILBOX # ***** 65 *****
003516 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003520 000000 HALT ;V NOT CLEARED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 745
761 003522 3\$:
003522 103005 BCC 4\$
003524 012737 000066 000302 MOV #66,\$FATAL ;MOVE TO MAILBOX # ***** 66 *****
003532 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003534 000000 HALT ;C NOT CLEARED
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 737
762 003536 032737 000340 177776 4\$: BIT #340,CC
763 003544 001405 BEQ 5\$
003546 012737 000067 000302 MOV #67,\$FATAL ;MOVE TO MAILBOX # ***** 67 *****
003554 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003556 000000 HALT ;PRIORITY NOT ZERO
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 726
764 003560 012706 000500 5\$: MOV #BUFF,SP
765 003564 012737 003602 000034 MOV #RETG1,RTRAP1
766 003572 012737 000357 000036 MOV #357,RTRAP1+2 ;SET NEW 'CC' AND PRIORITY
767 003600 104400 TRAP ;TRAP HERE
768 003602 RETG1:
769 003602 100405 BMI 1\$
003604 012737 000070 000302 MOV #70,\$FATAL ;MOVE TO MAILBOX # ***** 70 *****
003612 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
003614 000000 HALT ;N NOT SET
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 707

777

.SBTTL TEST #17 - TEST THAT ALL COMB "TRAP" WILL CAUSE A TRAP
:*****
:TEST 17 - TEST THAT ALL COMB "TRAP" WILL CAUSE A TRAP
:*****

003712	005237	000304		TST17:	INC	\$TESTN	;UPDATE TEST NUMBER
003716	022737	000017	000304		CMP	#17,\$TESTN	;SEQUENCE ERROR?
003724	001011				BNE	BR45	;BR TO ERROR HALT ON SEQ ERROR
778 003726	012737	104400	003746		MOV	#TRAP,RB1	;INITIALIZE BASE TRAP INSTRUCTION
779 003734	012737	003762	000034		MOV	#RA1,34	;RETURN FROM TRAP TO RA1
780 003742	012706	000500		RC1:	MOV	#BUFF,SP	;SET UP STACK POINTER
781 003746	104400			RB1:	TRAP		;TRAP INST WILL BE MODIFIED TO TRAP+377
782 003750				BR45:			
003750	012737	000075	000302		MOV	#75,\$FATAL	;MOVE TO MAILBOX # ***** 75 *****
003756	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
003760	000000				HALT		;PREVIOUS INST FAILED TO TRAP,OR WRONG \$TSTNM
							;TO SCOPE REPLACE HALT WITH 240
							;AND REPLACE NEXT INST WITH 761
783 003762	005237	003746		RA1:	INC	RB1	;INCREMENT TRAP INSTRUCTION
784 003766	022737	104777	003746		CMP	#104777,RB1	;TRAP+377 TO UPPER LIMIT
785 003774	103362				BHIS	RC1	;HAVE WE TESTED ALL
786 003776	012737	000036	000034		MOV	#36,34	
787 004004	005037	000036			CLR	36	

788

```

.SBTTL TEST #20 - TEST THAT A TRAP OCCURES ON AN 'IOT' INSTRUCTION
;*****
;TEST 20 - TEST THAT A TRAP OCCURES ON AN 'IOT' INSTRUCTION
;*****
TST20: INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #20,$TESTN ;SEQUENCE ERROR?
        BNE     TST21-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETA2,RTRAP2 ;RETURN LOCATION
        IOT     ;RESERVE INSTRUCTION, SHOULD TRAP
        MOV     #76,$FATAL  ;MOVE TO MAILBOX # ***** 76 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT    ;IOT DIDN'T TRAP,OR WRONG $STNM
                ;TO SCOPE REPLACE HALT WITH 240
                ;AND REPLACE NEXT INST WITH 764

```

```

004010 005237 000304
004014 022737 000020 000304
004022 001006
789 004024 012706 000500
790 004030 012737 004052 000020
791 004036 000004
792 004040 012737 000076 000302
004046 005212
004050 000000

```

793 004052

RETA2:

794

```

.SBTTL TEST #21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
TST21: INC $TESTN ;UPDATE TEST NUMBER
      CMP #21,$TESTN ;SEQUENCE ERROR?
      BNE TST22-12 ;BR TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #RETB2,RTRAP2 ;RETURN POINTER
      IOT ;RESERVED INSTRUCTION
RETB2: CMP SP,#BUFF-4 ;TEST DECREMENT OF SP
      BEQ TST22
      MOV #77,$FATAL ;MOVE TO MAILBOX # ***** 77 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;NOT DECREMENTED TWO WORDS,OR WRONG $TESTNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 761

```

```

004052 005237 000304
004056 022737 000021 000304
004064 001011
795 004066 012706 000500
796 004072 012737 004102 000020
797 004100 C00004
798 004102 020627 000474
799 004106 001405
004110 012737 000077 000302
004116 005212
004120 000000

```

800

.SBTTL TEST #22 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 22 - TEST THAT PROPER P.C. IS SAVED
:*****

004122	005237	000304		TST22:	INC	\$TESTN	:UPDATE TEST NUMBER
004126	022737	000022	000304		CMP	#22,\$TESTN	:SEQUENCE ERROR?
004134	001012				BNE	TST23-12	:BR TO ERROR HALT ON SEQ ERROR
801 004136	012706	000500			MOV	#BJFF,SP	:STACK POINTER SETUP
802 004142	012737	004152	000020		MOV	#RET2,RTRAP2	:RETURN FROM TRAP POINTER
803 004150	000004				IOT		:TRAP ON THIS INSTRUCTION
804 004152	022737	004152	000474	RET2:	CMP	#.,BUFF-4	:CHECK FOR INCREMENTED P.C.
805 004160	001405				BEQ	TST23	
004162	012737	000100	000302		MOV	#100,\$FATAL	:MOVE TO MAILBOX # ***** 100 *****
004170	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004172	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

821

.SBTTL TEST #24 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****
 :TEST 24 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****

004322	005237	000304			TST24:	INC	\$TESTN		:UPDATE TEST NUMBER
004326	022737	000024	000304			CMP	#24,\$TESTN		:SEQUENCE ERROR?
004334	001121					BNE	BR46		:BR TO ERROR HALT ON SEQ ERROR
822 004336	012706	000500				MOV	#BUFF,SP		
823 004342	012737	004356	000020			MOV	#RETG2,RTRAP2		
824 004350	005037	000022				CLR	RTRAP2+2		:CLEAR FUTURE PRIORITY AND CC
825 004354	000004					IOT			
826 004356					RETG2:				:TEST FOR 'C' CLEARED
827 004356	100005					BPL	1\$		
004360	012737	000103	000302			MOV	#103,\$FATAL		:MOVE TO MAILBOX # ***** 103 *****
004366	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004370	000000					HALT			:N NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 761
828 004372					1\$:				
004372	001005					BNE	2\$		
004374	012737	000104	000302			MOV	#104,\$FATAL		:MOVE TO MAILBOX # ***** 104 *****
004402	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004404	000000					HALT			:Z NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 753
829 004406					2\$:				
004406	102005					BVC	3\$		
004410	012737	000105	000302			MOV	#105,\$FATAL		:MOVE TO MAILBOX # ***** 105 *****
004416	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004420	000000					HALT			:V NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 745
830 004422					3\$:				
004422	103005					BCC	4\$		
004424	012737	000106	000302			MOV	#106,\$FATAL		:MOVE TO MAILBOX # ***** 106 *****
004432	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004434	000000					HALT			:C NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 737
831 004436	032737	000340	177776		4\$:	BIT	#340,CC		:TEST PRIORITY
832 004444	001405					BEQ	5\$		
004446	012737	000107	000302			MOV	#107,\$FATAL		:MOVE TO MAILBOX # ***** 107 *****
004454	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004456	000000					HALT			:PRIORITY NOT ZERO :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 726
833 004460	012706	000500			5\$:	MOV	#BUFF,SP		
834 004464	012737	004502	000020			MOV	#RETG2,RTRAP2		
835 004472	012737	000357	000022			MOV	#357,RTRAP2+2		:SET NEW 'CC' AND PRIORITY
836 004500	000004					IOT			:TRAP HERE
837 004502					RETG2:				
838 004502	100405					BMI	1\$		
004504	012737	000110	000302			MOV	#110,\$FATAL		:MOVE TO MAILBOX # ***** 110 *****
004512	005212					INC	(R2)		:SET MSGTYP TO FATAL ERROR
004514	000000					HALT			:N NOT SET :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 707

839	004516			1\$:	BEO	2\$			
	004516	001405			MOV	#111,\$FATAL		:MOVE TO MAILBOX # ***** 111 *****	
	004520	012737	000111	000302	INC	(R2)		:SET MSGTYP TO FATAL ERROR	
	004526	005212			HALT			:Z NOT SET	
	004530	000000						:TO SCOPE REPLACE HALT WITH 240	
								:AND REPLACE NEXT INST WITH 701	
840	004532			2\$:	BVS	3\$			
	004532	102405			MOV	#112,\$FATAL		:MOVE TO MAILBOX # ***** 112 *****	
	004534	012737	000112	000302	INC	(R2)		:SET MSGTYP TO FATAL ERROR	
	004542	005212			HALT			:V NOT SET	
	004544	000000						:TO SCOPE REPLACE HALT WITH 240	
								:AND REPLACE NEXT INST WITH 673	
841	004546			3\$:	BCS	4\$			
	004546	103405			MOV	#113,\$FATAL		:MOVE TO MAILBOX # ***** 113 *****	
	004550	012737	000113	000302	INC	(R2)		:SET MSGTYP TO FATAL ERROR	
	004556	005212			HALT			:C NOT SET	
	004560	000000						:TO SCOPE REPLACE HALT WITH 240	
								:AND REPLACE NEXT INST WITH 665	
842	004562	013706	177776	4\$:	MOV	CC,SP			
843	004566	042706	000017		BIC	#17,SP			
844	004572	022706	000340		CMP	#340,SP			
845	004576	001405			BEO	BR46A			
	004600			BR46:	MOV	#114,\$FATAL		:MOVE TO MAILBOX # ***** 114 *****	
	004600	012737	000114	000302	INC	(R2)		:SET MSGTYP TO FATAL ERROR	
	004606	005212			HALT			:PRIORITY WAS CHANGED,OR WRONG \$STNM	
	004610	000000						:TO SCOPE REPLACE HALT WITH 240	
								:AND REPLACE NEXT INST WITH 651	
846	004612	012737	000022	000020	BR46A: MOV	#22,20		:.+2	
847	004620	005037	000022		CLR	22		:HALT	

848

```

.SBTTL TEST #25 - TEST THAT A TRAP OCCURS ON AN EMT    INS
:*****
:TEST 25 - TEST THAT A TRAP OCCURS ON AN EMT    INS
:*****
TST25:  INC        $TESTN            ;UPDATE TEST NUMBER
         CMP        #25,$TESTN        ;SEQUENCE ERROR?
         BNE        TST26-12        ;BR TO ERROR HALT ON SEQ ERROR
         MOV        #BUFF,SP        ;STACK POINTER SETUP
         MOV        #RETA3,RTRAP3    ;RETURN LOCATION
         EMT                       ;RESERVE INSTRUCTION, SHOULD TRAP
         MOV        #115,$FATAL      ;MOVE TO MAILBOX # ***** 115 *****
         INC        (R2)            ;SET MSGTYP TO FATAL ERROR
         HALT                      ;EMT DIDN'T TRAP,OR WRONG $TSTNM
                                    ;TO SCOPE REPLACE HALT WITH 240
                                    ;AND REPLACE NEXT INST WITH 764
  
```

```

004624 005237 000304
004630 022737 000025 000304
004636 001006
849 004640 012706 00050C
850 004644 012737 004666 000030
851 004652 104000
852 004654 012737 000115 000302
004662 005212
004664 000000
  
```

853 004666

RETA3:

854

```

.SBTTL TEST #26 - TEST DEC OF STACK POINTER ON A TRAP OPER
:*****
:TEST 26 - TEST DEC OF STACK POINTER ON A TRAP OPER
:*****
TST26:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #26,$TESTN  ;SEQUENCE ERROR?
        BNE     TST27-12     ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP     ;STACK POINTER SETUP
        MOV     #RETB3,RTRAP3 ;RETURN POINTER
        EMT                      ;RESERVED INSTRUCTION
RETB3:  CMP     SP,#BUFF-4    ;TEST DECREMENT OF SP
        BEQ     TST27
        MOV     #116,$FATAL   ;MOVE TO MAILBOX # ***** 116 *****
        INC     (R2)          ;SET MSGTYP TO FATAL ERROR
        HALT                    ;NOT DECREMENTED TWO WORDS,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 761
  
```

```

004666 005237 000304
004672 022737 000026 000304
004700 001011
855 004702 012706 000500
856 004706 012737 004716 000030
857 004714 104000
858 004716 020627 000474
859 004722 001405
    004724 012737 000116 000302
    004732 005212
    004734 000000
  
```

860

```

.SBTTL TEST #27 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 27 - TEST THAT PROPER P.C. IS SAVED
:*****
TST27: INC $TESTN ;UPDATE TEST NUMBER
      CMP #27,$TESTN ;SEQUENCE ERROR?
      BNE TST30-12 ;BR TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #RETC3,RTRAP3 ;RETURN FROM TRAP POINTER
      EMT ;TRAP ON THIS INSTRUCTION
RETC3: CMP #,$BUFF-4 ;CHECK FOR INCREMENTED P.C.
      BEQ TST30
      MOV #117,$FATAL ;MOVE TO MAILBOX # ***** 117 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT P.C.,OR WRONG $STNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 760

```

```

004736 005237 000304
004742 022737 000027 000304
004750 001012
861 004752 012706 000500
862 004756 012737 004766 000030
863 004764 104000
864 004766 022737 004766 000474
865 004774 001405
004776 012737 000117 000302
005004 005212
005006 000000

```


866

```

.SBTTL TEST #30 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
*****
;TEST 30 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
*****
TST30:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #30,$TESTN  ;SEQUENCE ERROR?
        BNE      TST31-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV      #BUFF,SP    ;SET UP
        MOV      #RETD3,RTRAP3 ;SET UP
        CLR      CC          ;CLEAR CC AND PRIORITY

        EMT                    ;TRAP
RETD3:  CMP      BUFF-2,#0    ;TEST THAT OLD STATUS WENT TO STACK
        BEQ      1$
        MOV      #120,$FATAL  ;MOVE TO MAILBOX # ***** 120 *****
        INC      (R2)         ;SET MSGTYP TO FATAL ERROR
        HALT                ;INCORRECT STATUS
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 755

1$:     MOV      #BUFF,SP    ;SET UP
        MOV      #RETE3,RTRAP3 ;SET UP
        MOV      #357,CC     ;SET PRIORITY
        SCC                    ;SET CC
        EMT                    ;TRAP
RETE3:  CMP      BUFF-2,#357  ;COMPARES STATUS ON STACK
        BEQ      TST31
        MOV      #121,$FATAL  ;MOVE TO MAILBOX # ***** 121 *****
        INC      (R2)         ;SET MSGTYP TO FATAL ERROR
        HALT                ;INCORRECT STATUS ON STACK,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 732

```

```

005010 005237 000304
005014 022737 000030 000304
005022 001040
867 005024 012706 000500
868 005030 012737 005046 000030
869 005036 005037 177776
870 005042 000257
871 005044 104000
872 005046 023727 000476 000000
873 005054 001405
    005056 012737 000120 000302
    005064 005212
    005066 000000

874 005070 012706 000500
875 005074 012737 005114 000030
876 005102 012737 000357 177776
877 005110 000277
878 005112 104000
879 005114 023727 000476 000357
880 005122 001405
    005124 012737 000121 000302
    005132 005212
    005134 000000

```

TEST #31 - TEST THAT 'NEW' STATUS IS CORRECT

D 4

.SBTTL TEST #31 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****
 :TEST 31 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****

```

005136 005237 000304
005142 022737 000031 000304
005150 001117
882 005152 012706 000500
883 005156 012737 005172 000030
884 005164 005037 000032
885 005170 104000
886 005172
887 005172 100005
005174 012737 000122 000302
005202 005212
005204 000000

TST31: INC $TESTN ;UPDATE TEST NUMBER
        CMP #31,$TESTN ;SEQUENCE ERROR?
        BNE TST32-12 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,SP
        MOV #RETG3,RTRAP3
        CLR RTRAP3+2 ;CLEAR FUTURE PRIORITY AND CC
        EMT

RET3: ;TEST FOR 'C' CLEARED

1$: BPL 1$
    MOV #122,$FATAL ;MOVE TO MAILBOX # ***** 122 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761

888 005206 001005 1$: BNE 2$
    MOV #123,$FATAL ;MOVE TO MAILBOX # ***** 123 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;Z NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 753

889 005222 102005 2$: BVC 3$
    MOV #124,$FATAL ;MOVE TO MAILBOX # ***** 124 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;V NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 745

890 005236 103005 3$: BCC 4$
    MOV #125,$FATAL ;MOVE TO MAILBOX # ***** 125 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 737

891 005252 032737 000340 177776 4$: BIT #340,CC
892 005260 001405 5$: BEQ 5$
    MOV #126,$FATAL ;MOVE TO MAILBOX # ***** 126 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;PRIORITY NOT ZERO
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 726

893 005274 012706 000500 5$: MOV #BUFF,SP
894 005300 012737 005316 000030 MOV #RETG3,RTRAP3
895 005306 012737 000357 000032 MOV #357,RTRAP3+2
896 005314 104000 EMT ;SET NEW 'CC' AND PRIORITY
897 005316 RETG3: ;TRAP HERE
898 005316 100405 BMI 1$
    MOV #127,$FATAL ;MOVE TO MAILBOX # ***** 127 *****
    INC (R2) ;SET MSGTYP TO FATAL ERROR
    HALT ;N NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 707
    
```


905

```

.SBTTL TEST #32 - TEST THAT ALL COMB EMT WILL CAUSE A TRAP
:*****
:TEST 32 - TEST THAT ALL COMB EMT WILL CAUSE A TRAP
:*****
TST32:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #32,$TESTN ;SEQUENCE ERROR?
        BNE      BR47       ;BR TO ERROR HALT ON SEQ ERROR
        MOV      #EMT,RB    ;INITIALIZE BASE EMT INSTRUCTION
        MOV      #RA,30     ;RETURN FROM TRAP TO RA
RC:     MOV      #BUFF,SP   ;SET UP STACK POINTER
RB:     EMT
BR47:
        MOV      #134,$FATAL ;MOVE TO MAILBOX # ***** 134 *****
        INC      (R2)       ;SET MSGTYP TO FATAL ERROR
        HALT              ;PREVIOUS INST FAILED TO TRAP,OR WRONG $STNM
                          ;TO SCOPE REPLACE HALT WITH 240
                          ;AND REPLACE NEXT INST WITH 761
RA:     INC      RB        ;INCREMENT TRAP INSTRUCTION
        CMP      #104377,RB ;EMT+377 TO EMT?
        BHIS    RC         ;HAVE WE TESTED ALL
        MOV      #32,30    ;YES
        CLR     32        ;/.+
                          ;HALT
    
```

```

005422 005237 000304
005426 022737 000032 000304
005434 001011
906 005436 012737 104000 005456
907 005444 012737 005472 000030
908 005452 012706 000500
909 005456 104000
910 005460
    005460 012737 000134 000302
    005466 005212
    005470 000000

911 005472 005237 005456
912 005476 022737 104377 005456
913 005504 103362
914
915 005506 012737 000032 000030
916 005514 005037 000032
    
```

917

```
.SBTTL TEST #33 - TEST THAT A TRAP OCCURES ON AN "TRACE-TRT" INS
:*****
:TEST 33 - TEST THAT A TRAP OCCURES ON AN "TRACE-TRT" INS
:*****
TST33:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #33,$TESTN  ;SEQUENCE ERROR?
        BNE     TST34-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETA4,RTRAP4 ;RETURN LOCATION
        TRT     ;RESERVED INSTRUCTION, SHOULD TRAP
        MOV     #135,$FATAL  ;MOVE TO MAILBOX # ***** 135 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT    ;TRT DIDN'T TRAP,OR WRONG $STNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 764
```

```
005520 005237 000304
005524 022737 000033 000304
005532 001006
918 005534 012706 000500
919 005540 012737 005562 000014
920 005546 000003
921 005550 012737 000135 000302
    005556 005212
    005560 000000
```

922 005562

RETA4:

923

```

.SBTTL TEST #34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
TST34:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #34,$TESTN ;SEQUENCE ERROR?
        BNE     TST35-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RET84,RTRAP4 ;RETURN POINTER
        TRT
RET84:  CMP     SP,#BUFF-4   ;RESERVED INSTRUCTION
        BEQ     TST35       ;TEST DECREMENT OF SP
        MOV     #136,$FATAL ;MOVE TO MAILBOX # ***** 136 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT              ;NOT DECREMENTED TWO WORDS,OR WRONG $TSTNM
                          ;TO SCOPE REPLACE HALT WITH 240
                          ;AND REPLACE NEXT INST WITH 761
  
```

```

005562 005237 000304
005566 022737 000034 000304
005574 001011
924 005576 012706 000500
925 005602 012737 005612 000014
926 005610 000003
927 005612 020627 000474
928 005616 001405
    005620 012737 000136 000302
    005626 005212
    005630 000000
  
```

929

```
.SBTTL TEST #35 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 35 - TEST THAT PROPER P.C. IS SAVED
:*****
TST35: INC $TESTN ;UPDATE TEST NUMBER
      CMP #35,$TESTN ;SEQUENCE ERROR?
      BNE TST36-12 ;BR TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #RETC4,RTRAP4 ;RETURN FROM TRAP POINTER
      TRT ;TRAP ON THIS INSTRUCTION
RETC4: CMP #.,BUFF-4 ;CHECK FOR INCREMENTED P.C.
      BEQ TST36
      MOV #137,$FATAL ;MOVE TO MAILBOX # ***** 137 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT P.C.,OR WRONG $STNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 760
```

```
005632 005237 000304
005636 022737 000035 000304
005644 001012
930 005646 012706 000500
931 005652 012737 005662 000014
932 005660 000003
933 005662 022737 005662 000474
934 005670 001405
005672 012737 000137 000302
005700 005212
005702 000000
```

935

```
.SBTTL TEST #36 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
:TEST 36 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
TST36: INC $TESTN ;UPDATE TEST NUMBER
      CMP #36,$TESTN ;SEQUENCE ERROR?
      BNE TST37-12 ;BR TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;SET UP
      MOV #RETD4,RTRAP4 ;SET UP
      CLR CC ;CLEAR CC AND PRIORITY
      TRT ;TRAP
RETD4: CMP BUFF-2,#0 ;TEST THAT OLD STATUS WENT TO STACK
      BEQ 1$ ;TEST FOR ALL ZEROS
      MOV #140,$FATAL ;MOVE TO MAILBOX # ***** 140 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT STATUS
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 755
1$: MOV #BUFF,SP ;SET UP
     MOV #RETE4,RTRAP4 ;SET UP
     MOV #357,CC ;SET PRIORITY
     SCC ;SET-SET CC
     TRT ;TRAP
RETE4: CMP BUFF-2,#357 ;COMPARES STATUS ON STACK
      BEQ TST37
      MOV #141,$FATAL ;MOVE TO MAILBOX # ***** 141 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT STATUS ON STACK,OR WRONG $STNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 732
```

```
005704 005237 000304
005710 022737 000036 000304
005716 001040
936 005720 012706 000500
937 005724 012737 005742 000014
938 005732 005037 177776
939 005736 000257
940 005740 000003
941 005742 023727 000476 000000
942
943 005750 001405
     005752 012737 000140 000302
     005760 005212
     005762 000000

944 005764 012706 000500
945 005770 012737 006010 000014
946 005776 012737 000357 177776
947 006004 000277
948 006006 000003
949 006010 023727 000476 000357
950 006016 001405
     006020 012737 000141 000302
     006026 005212
     006030 000000
```


951

.SBTTL TEST #37 - TEST THAT 'NEW' STATUS IS CORRECT

 ;TEST 37 - TEST THAT 'NEW' STATUS IS CORRECT

```

006032 005237 000304
006036 022737 000037 000304
006044 001121
952 006046 012706 000500
953 006052 012737 006066 000014
954 006060 005037 000016
955 006064 000003
956 006066
957 006066 100005
006070 012737 000142 000302
006076 005212
006100 000000

TST37: INC $TESTN ;UPDATE TEST NUMBER
        CMP #37,$TESTN ;SEQUENCE ERROR?
        BNE BR51 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,SP
        MOV #RETF4,RTRAP4
        CLR RTRAP4+2 ;CLEAR FUTURE PRIORITY AND CC
        TRT

RETF4: ;TEST FOR 'C' CLEARED
        BPL 1$
        MOV #142,$FATAL ;MOVE TO MAILBOX # ***** 142 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761

1$:
        BNE 2$
        MOV #143,$FATAL ;MOVE TO MAILBOX # ***** 143 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;Z NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 753

2$:
        BVC 3$
        MOV #144,$FATAL ;MOVE TO MAILBOX # ***** 144 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;V NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 745

3$:
        BCC 4$
        MOV #145,$FATAL ;MOVE TO MAILBOX # ***** 145 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 737

4$:
        BIT #340,CC
        BEQ 5$
        MOV #146,$FATAL ;MOVE TO MAILBOX # ***** 146 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;PRIORITY NOT ZERO
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 726

5$:
        MOV #BUFF,SP
        MOV #RETF4,RTRAP4
        MOV #357,RTRAP4+2 ;SET NEW 'CC' AND PRIORITY
        TRT ;TRAP HERE

RETF4:
        BMI 1$
        MOV #147,$FATAL ;MOVE TO MAILBOX # ***** 147 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;N NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 707
    
```

```

969 006226          1$:
    006226 001405          BEQ      2$
    006230 012737 000150 000302  MOV     #150,$FATAL      ;MOVE TO MAILBOX # ***** 150 *****
    006236 005212          INC     (R2)              ;SET MSGTYP TO FATAL ERROR
    006240 000000          HALT                    ;Z NOT SET
                                           ;TO SCOPE REPLACE HALT WITH 240
                                           ;AND REPLACE NEXT INST WITH 701

970 006242          2$:
    006242 102405          BVS     3$
    006244 012737 000151 000302  MOV     #151,$FATAL      ;MOVE TO MAILBOX # ***** 151 *****
    006252 005212          INC     (R2)              ;SET MSGTYP TO FATAL ERROR
    006254 000000          HALT                    ;V NOT SET
                                           ;TO SCOPE REPLACE HALT WITH 240
                                           ;AND REPLACE NEXT INST WITH 673

971 006256          3$:
    006256 103405          BCS     4$
    006260 012737 000152 000302  MOV     #152,$FATAL      ;MOVE TO MAILBOX # ***** 152 *****
    006266 005212          INC     (R2)              ;SET MSGTYP TO FATAL ERROR
    006270 000000          HALT                    ;C NOT SET
                                           ;TO SCOPE REPLACE HALT WITH 240
                                           ;AND REPLACE NEXT INST WITH 665

972 006272 013706 177776          4$:  MOV     CC,SP
973 006276 042706 000017          BIC     #17,SP
974 006302 022706 000340          CMP     #340,SP
975 006306 001405          BEQ     BR51A
    006310          BR51:  MOV     #153,$FATAL      ;MOVE TO MAILBOX # ***** 153 *****
    006310 012737 000153 000302  INC     (R2)              ;SET MSGTYP TO FATAL ERROR
    006316 005212          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
    006320 000000          ;TO SCOPE REPLACE HALT WITH 240
                                           ;AND REPLACE NEXT INST WITH 651

976 006322 012737 000016 000014 BR51A: MOV     #16,14
977 006330 005037 000016          CLR     16
978
979          ;PDP-11 ILLEGAL AND ADDRESS INSTRUCTION TEST
980          ;ALL INSTRUCTIONS THAT ARE ILLEGAL
981          ;SHOULD TRAP TO LOCATION 10, AND THE
982          ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
983          ;SHOULD BE PLACED ON THE STACK
984
  
```

985

.SBTTL TEST #40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
 :*****
 :TEST 40 - TEST THAT A TRAP OCCURS ON AN ILLEGAL INS
 :*****

006334 005237 000304
 006340 022737 000040 0C0304
 006346 001006
 986 006350 012706 000500
 987 006354 012737 006376 000010
 988 006362 0C0100
 989 006364 012737 000154 000302
 006372 005212
 006374 000000

TST40: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #40,\$TESTN ;SEQUENCE ERROR?
 BNE TST41-12 ;BR TO ERROR HALT ON SEQ ERROR
 MOV #BUFF,SP ;STACK POINTER SETUP
 MOV #RETA5,RTRAP ;RETURN LOCATION
 JMP %0 ;ILLEGAL INSTRUCTION, SHOULD TRAP
 MOV #154,\$FATAL ;MOVE TO MAILBOX # ***** 154 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;ILLEGAL INSTRUCTION DIDN'T TRAP,OR WRONG \$TESTN
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 764

990 006376

RETA5:

997

```
.SBTTL TEST #42 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 42 - TEST THAT PROPER P.C. IS SAVED
:*****
TST42: INC $TESTN ;UPDATE TEST NUMBER
      CMP #42,$TESTN ;SEQUENCE ERROR?
      BNE TST43-12 ;BR TO ERROR HALT ON SEG ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #RETC5,RTRAP ;RETURN FROM TRAP POINTER
      JMP %0 ;TRAP ON THIS INSTRUCTION
RETC5: CMP #.,BUFF-4 ;CHECK FOR INCREMENTED P.C.
      BEQ TST43
      MOV #156,$FATAL ;MOVE TO MAILBOX # ***** 156 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT P.C.,OR WRONG $STNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 760
```

```
006446 005237 000304
006452 022737 000042 000304
006460 001012
998 006462 012706 000500
999 006466 012737 006476 000010
1000 006474 000100
1001 006476 022737 006476 000474
1002 006504 001405
      006506 012737 000156 000302
      006514 005212
      006516 000000
```

1003

```

.SBTTL TEST #43 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
:TEST 43 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
TST43: INC $TESTN ;UPDATE TEST NUMBER
        CMP #43,$TESTN ;SEQUENCE ERROR?
        BNE TST44-12 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,SP ;SET UP
        MOV #RETD5,RTRAP ;SET UP
        CLR CC ;CLEAR CC AND PRIORITY
        JMP %0 ;TRAP
RETDS:  CMP BUFF-2,#0 ;TEST THAT OLD STATUS WENT TO STACK
        BEQ 1$
        MOV #157,$FATAL ;MOVE TO MAILBOX # ***** 157 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;INCORRECT STATUS
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 755
1$: MOV #BUFF,SP ;SET UP
    MOV #RETE5,RTRAP ;SET UP
    MOV #357,CC ;SET PRIORITY
    SCC ;SET CC
    JMP %0 ;TRAP
RETE5: CMP BUFF-2,#357 ;COMPARES STATUS ON STACK
        BEQ TST44
        MOV #160,$FATAL ;MOVE TO MAILBOX # ***** 160 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;INCORRECT STATUS ON STACK,OR WRONG $TESTNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 732
    
```

```

006520 005237 000304
006524 022737 000043 000304
006532 001040
1004 006534 012706 000500
1005 006540 012737 006556 000010
1006 006546 005037 177776
1007 006552 000257
1008 006554 000100
1009 006556 023727 000476 000000
1010 006564 001405
    006566 012737 000157 000302
    006574 005212
    006576 000000

1011 006600 012706 000500
1012 006604 012737 006624 000010
1013 006612 012737 000357 177776
1014 006620 000277
1015 006622 000100
1016 006624 023727 000476 000357
1017 006632 001405
    006634 012737 000160 000302
    006642 005212
    006644 000000
    
```

1018

.SBTTL TEST #44 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****
 :TEST 44 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****

```

006646 005237 000304
006652 022737 000044 000304
006660 001117
1019 006662 012706 000500
1020 006666 012737 006702 000010
1021 006674 005037 000012
1022 006700 000100
1023 006702
1024 006702 100005
006704 012737 000161 000302
006712 005212
006714 000000

TST44: INC $TESTN ;UPDATE TEST NUMBER
        CMP #44,$TESTN ;SEQUENCE ERROR?
        BNE TST45-12 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,SP
        MOV #RETF5,RTRAP
        CLR RTRAP+2 ;CLEAR FUTURE PRIORITY AND CC
        JMP %0

RETF5: ;TEST FOR 'C' CLEARED
        BPL 1$
        MOV #161,$FATAL ;MOVE TO MAILBOX # ***** 161 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761

1025 006716 1$:
006716 001005
006720 012737 000162 000302
006726 005212
006730 000000
        BNC 2$
        MOV #162,$FATAL ;MOVE TO MAILBOX # ***** 162 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;Z NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 753

1026 006732 2$:
006732 102005
006734 012737 000163 000302
006742 005212
006744 000000
        BVC 3$
        MOV #163,$FATAL ;MOVE TO MAILBOX # ***** 163 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;V NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 745

1027 006746 3$:
006746 103005
006750 012737 000164 000302
006756 005212
006760 000000
        BCC 4$
        MOV #164,$FATAL ;MOVE TO MAILBOX # ***** 164 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;C NOT CLEARED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 737

1028 006762 032737 000357 177776 4$:
1029 006770 001405
006772 012737 000165 000302
007000 005212
007002 000000
        BIT #357,CC
        BEQ 5$
        MOV #165,$FATAL ;MOVE TO MAILBOX # ***** 165 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;PRIORITY NOT ZERO
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 726

1030 007004 012706 000500
1031 007010 012737 007026 000010
1032 007016 012737 000357 000012
1033 007024 000100
1034 007026
1035 007026 100405
007030 012737 000166 000302
007036 005212
007040 000000

5$: MOV #BUFF,SP
    MOV #RETF5,RTRAP
    MOV #357,RTRAP+2 ;SET NEW 'CC' AND PRIORITY
    JMP %0 ;TRAP HERE

RETF5: BMI 1$
        MOV #166,$FATAL ;MOVE TO MAILBOX # ***** 166 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;N NOT SET
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 707
  
```

1036	007042			1\$:				
	007042	001405			BEQ	2\$		
	007044	012737	000167	000302	MOV	#167,\$FATAL	:MOVE TO MAILBOX # ***** 167 *****	
	007052	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	007054	000000			HALT		:Z NOT SET	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 701	
1037	007056			2\$:				
	007056	102405			BVS	3\$		
	007060	012737	000170	000302	MOV	#170,\$FATAL	:MOVE TO MAILBOX # ***** 170 *****	
	007066	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	007070	000000			HALT		:V NOT SET	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 673	
1038	007072			3\$:				
	007072	103405			BCS	4\$		
	007074	012737	000171	000302	MOV	#171,\$FATAL	:MOVE TO MAILBOX # ***** 171 *****	
	007102	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	007104	000000			HALT		:C NOT SET	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 665	
1039	007106	013706	177776	4\$:	MOV	CC,SP		
1040	007112	022706	000357		CMP	#357,SP		
1041	007116	001405			BEQ	TST45		
	007120	012737	000172	000302	MOV	#172,\$FATAL	:MOVE TO MAILBOX # ***** 172 *****	
	007126	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	007130	000000			HALT		:PRIORITY WAS CHANGED,OR WRONG \$STNM	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 653	

1042

```
.SBTTL TEST #45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
:*****
:TEST 45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
:*****
TST45: INC      $TESTN      ;UPDATE TEST NUMBER
      CMP      #45,$TESTN ;SEQUENCE ERROR?
      BNE     TST46-12    ;BR TO ERROR HALT ON SEQ ERROR
      MOV     #BUFF,SP    ;STACK POINTER SETUP
      MOV     #RETH5,RTRAP;RETURN LOCATION
      JSR     %0,%0       ;RESERVED INS; SHOULD TRAP
      MOV     #173,$FATAL ;MOVE TO MAILBOX # ***** 173 *****
      INC     (R2)        ;SET MSGTYP TO FATAL ERROR
      HALT                ;DIDN'T TRAP,OR WRONG $STNM
                        ;TO SCOPE REPLACE HALT WITH 240
                        ;AND REPLACE NEXT INST WITH 764
```

```
007132 005237 000304
007136 022737 000045 000304
007144 001006
1043 007146 012706 000500
1044 007152 012737 007174 000010
1045 007160 004000
1046 007162 012737 000173 000302
007170 005212
007172 000000
```

1047 007174

RETH5:

1048

```

.SBTTL TEST #46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
TST46:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #46,$TESTN ;SEQUENCE ERROR?
        BNE     TST47-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETJ,RTRAP ;RETURN POINTER
        JSR     %0,%0       ;RESERVED INS
RETJ:   CMP     SP,#BUFF-4  ;TEST DECREMENT OF SP
        BEQ     TST47
        MOV     #174,$FATAL ;MOVE TO MAILBOX # ***** 174 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;NOT DECREMENTED TWO WORDS,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 761
  
```

```

007174 005237 000304
007200 022737 000046 000304
007206 001011
1049 007210 012706 000500
1050 007214 012737 007224 000010
1051 007222 004000
1052 007224 020627 000474
1053 007230 001405
    007232 012737 000174 000302
    007240 005212
    007242 000000
  
```

1054

```

.SBTTL TEST #47 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 47 - TEST THAT PROPER P.C. IS SAVED
:*****
TST47:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #47,$TESTN ;SEQUENCE ERROR?
        BNE     TST50-12    ;BR TO ERROR HALT ON SEQ ERROR
1055   007244  005237  000304  000304  ;STACK POINTER SETUP
1056   007250  022737  000047  000304  ;RETURN FROM TRAP POINTER
1057   007256  001012  000500  000010  ;TRAP ON THIS INS
1058   007260  012706  000500  000010  ;CHECK FOR INCREMENTED P.C.
1059   007264  012737  007274  000010  ;
1059   007272  004000  000000  000000  ;
1059   007274  022737  007274  000474  ;
1059   007302  001405  000175  000302  ;
1059   007304  012737  000175  000302  ;
1059   007312  005212  000000  000000  ;
1059   007314  000000  000000  000000  ;
INSTK:  JSR     %0,%0
RETK:   CMP     #INSTK+2,BUFF-4
        BEQ     TST50
        MOV     #175,$FATAL ;MOVE TO MAILBOX # ***** 175 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                    ;INCORRECT P.C.,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 760
  
```

1060

1061

```

.SBTTL TEST #50 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
:TEST 50 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
TST50:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #50,$TESTN ;SEQUENCE ERROR?
        BNE     TST51-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;SET UP
        MOV     #RETL,RTRAP ;SET UP
        CLR     CC          ;CLEAR CC AND PRIORITY
        CCC
        JSR     %0,%0       ;TRAP
RETL:   CMP     BUFF-2,#0   ;TEST THAT OLD STATUS WENT TO STACK
        BEQ     1$
        MOV     #176,$FATAL ;MOVE TO MAILBOX # ***** 176 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT               ;INCORRCT STATUS
                           ;TO SCOPE REPLACE HALT WITH 240
                           ;AND REPLACE NEXT INST WITH 755
1$:     MOV     #BUFF,SP    ;SET UP
        MOV     #RETM,RTRAP ;SET UP
        MOV     #357,CC     ;SET PRIORITY
        SCC
        JSR     %0,%0       ;TRAP
    
```

```

007316 005237 000304
007322 022737 000050 000304
007330 001040
1062 007332 012706 000500
1063 007336 012737 007354 000010
1064 007344 005037 177776
1065 007350 000257
1066 007352 004000
1067 007354 023727 000476 000000
1068 007362 001405
        007364 012737 000176 000302
        007372 005212
        007374 000000

1069 007376 012706 000500
1070 007402 012737 007422 000010
1071 007410 012737 000357 177776
1072 007416 000277
1073 007420 004000
    
```


.SBTTL TEST #51 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****
 :TEST 51 - TEST THAT 'NEW' STATUS IS CORRECT
 :*****

007444	005237	000304			TST51:	INC	\$TESTN	;UPDATE TEST NUMBER
007450	022737	000051	000304			CMP	#51,\$TESTN	;SEQUENCE ERROR?
007456	001116					BNE	TST52-12	;BR TO ERROR HALT ON SEQ ERROR
1078 007460	012706	000500				MOV	#BUFF,SP	
1079 007464	012737	007500	000010			MOV	#RETN,RTRAP	
1080 007472	005037	000012				CLR	RTRAP+2	;CLEAR FUTURE PRIORITY AND CC
1081 007476	004000					JSR	%0,%0	
1082 007500					RETN:			;TEST FOR 'C' CLEARED
1083 007500	100005					BPL	1\$	
	007502	012737	000200	000302		MOV	#200,\$FATAL	;MOVE TO MAILBOX # ***** 200 *****
	007510	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007512	000000				HALT		;C NOT CLEARED ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 761
1084 007514					1\$:			
	007514	001005				BNE	2\$	
	007516	012737	000201	000302		MOV	#201,\$FATAL	;MOVE TO MAILBOX # ***** 201 *****
	007524	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007526	000000				HALT		;Z NOT CLEARED ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 753
1085 007530					2\$:			
	007530	102005				BVC	3\$	
	007532	012737	000202	000302		MOV	#202,\$FATAL	;MOVE TO MAILBOX # ***** 202 *****
	007540	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007542	000000				HALT		;V NOT CLEARED ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 745
1086 007544					3\$:			
	007544	103005				BCC	4\$	
	007546	012737	000203	000302		MOV	#203,\$FATAL	;MOVE TO MAILBOX # ***** 203 *****
	007554	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007556	000000				HALT		;C NOT CLEARED ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 737
1087 007560	013700	177776			4\$:	MOV	CC,%0	;TEMP STORAGE
1088 007564	001405					BEQ	5\$	
	007566	012737	000204	000302		MOV	#204,\$FATAL	;MOVE TO MAILBOX # ***** 204 *****
	007574	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007576	000000				HALT		;PRIORITY NOT ZERO ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 727
1089 007600	012706	000500			5\$:	MOV	#BUFF,SP	
1090 007604	012737	007622	000010			MOV	#RETO,RTRAP	
1091 007612	012737	000357	000012			MOV	#357,RTRAP+2	;SET NEW 'CC' AND PRIORITY
1092 007620	004000					JSR	%0,%0	;TRAP HERE
1093 007622					RETO:			
1094 007622	100405					BMI	1\$	
	007624	012737	000205	000302		MOV	#205,\$FATAL	;MOVE TO MAILBOX # ***** 205 *****
	007632	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	007634	000000				HALT		;N NOT SET ;TO SCOPE REPLACE HALT WITH 240 ;AND REPLACE NEXT INST WITH 710

```

1095 007636          1$:
      007636 001405          BEQ      2$
      007640 012737 000206 000302      MOV      #206,$FATAL      ;MOVE TO MAILBOX # ***** 206 *****
      007646 005212          INC      (R2)      ;SET MSGTYP TO FATAL ERROR
      007650 000000          HALT          ;Z NOT SET
                                          ;TO SCOPE REPLACE HALT WITH 240
                                          ;AND REPLACE NEXT INST WITH 702

1096 007652          2$:
      007652 102405          BVS      3$
      007654 012737 000207 000302      MOV      #207,$FATAL      ;MOVE TO MAILBOX # ***** 207 *****
      007662 005212          INC      (R2)      ;SET MSGTYP TO FATAL ERROR
      007664 000000          HALT          ;V NOT SET
                                          ;TO SCOPE REPLACE HALT WITH 240
                                          ;AND REPLACE NEXT INST WITH 674

1097 007666          3$:
      007666 103405          BCS      4$
      007670 012737 000210 000302      MOV      #210,$FATAL      ;MOVE TO MAILBOX # ***** 210 *****
      007676 005212          INC      (R2)      ;SET MSGTYP TO FATAL ERROR
      007700 000000          HALT          ;C NOT SET
                                          ;TO SCOPE REPLACE HALT WITH 240
                                          ;AND REPLACE NEXT INST WITH 666

1098 007702 013700 177776          4$:
1099 007706 022700 000357          MOV      CC,$0
1100 007712 001405          CMP      #357,$0
      007714 012737 000211 000302      BEQ      TST52
      007722 005212          MOV      #211,$FATAL      ;MOVE TO MAILBOX # ***** 211 *****
      007724 000000          INC      (R2)      ;SET MSGTYP TO FATAL ERROR
                                          ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
                                          ;TO SCOPE REPLACE HALT WITH 240
                                          ;AND REPLACE NEXT INST WITH 654

1101
  
```

1102

.SBTTL TEST #52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
 :*****
 :TEST 52 - TEST THAT A TRAP OCCURES ON AN ILLEGAL ADDRESS
 :*****

007726 005237 000304
 007732 022737 000052 000304
 007740 001040
 1103 007742 005037 177766
 1104 007746 005737 177766
 1105 007752 001405
 007754 012737 000212 000302
 007762 005212
 007764 000000

TST52: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #52,\$TESTN ;SEQUENCE ERROR?
 BNE ERRP1 ;BR TO ERROR HALT ON SEQ ERROR
 CLR CPUERR ;CLEAR CPU ERROR REGISTER
 TST CPUERR ;VERIFY THAT IT CLEARED
 BEQ 1\$
 MOV #212,\$FATAL ;MOVE TO MAILBOX # ***** 212 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;CPU ERROR REG FAILED TO CLEAR
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 765

1106 007766 012706 000500
 1107 007772 012737 010016 000004
 1108 010000 005737 000001
 1109 010004 012737 000213 000302
 010012 005212
 010014 000000

1\$: MOV #BUFF,SP ;STACK POINTER SETUP
 MOV #RETP,RTRAPS ;RETURN LOCATION
 TST 1 ;ILLEGAL ADDRESS INS, SHOULD TRAP
 MOV #213,\$FATAL ;MOVE TO MAILBOX # ***** 213 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;ILLEGAL ADDRESS DID NOT TRAP
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 751

1110 010016
 1111 010016 013737 177766 000502
 1112 010024 042737 177413 000502
 1113 010032 022737 000100 000502
 1114 010040 001405
 010042
 010042 012737 000214 000302
 010050 005212
 010052 000000

RETP: MOV CPUERR,RCPUER ;READ AND SAVE CPU ERROR REGISTER
 BIC #CERMSK,RCPUER ;MASK OFF UNUSED BITS OF CPU ERROR REG
 CMP #100,RCPUER ;ODD ADDRESS BIT SET?
 BEQ CERR1
 ERRP1: MOV #214,\$FATAL ;MOVE TO MAILBOX # ***** 214 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERROR REG CONTENTS, OR WRONG \$STNM
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 732

1115 010054 005037 177766
 1116

CERR1: CLR CPUERR ;CLEAR ODD ADDRESS BIT

1117

```

.SBTTL TEST #53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
:TEST 53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
:*****
TST53:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #53,$TESTN ;SEQUENCE ERROR?
        BNE     TST54-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETQ,RTRAP5 ;RETURN POINTER
        TST    1           ;RESERVED INS
RETQ:   CMP     SP,#BUFF-4  ;TEST DECREMENT OF SP
        BEQ     TST54
        MOV     #215,$FATAL ;MOVE TO MAILBOX # ***** 215 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;NOT DECREMENTED TWO WORDS,OR WRONG $TESTNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 760
  
```

```

010060 005237 000304
010064 022737 000053 000304
010072 001012
1118 010074 012706 000500
1119 010100 012737 010112 000004
1120 010106 005737 000001
1121 010112 020627 000474
1122 010116 001405
    010120 012737 000215 000302
    010126 005212
    010130 000000
  
```

1123

```

.SBTTL TEST #54 - TEST THAT PROPER P.C. IS SAVED
:*****
:TEST 54 - TEST THAT PROPER P.C. IS SAVED
:*****
TST54:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #54,$TESTN ;SEQUENCE ERROR?
        BNE     TST55-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;STACK POINTER SETUP
        MOV     #RETR,RTRAPS;RETURN FROM TRAP POINTER
        TST    1           ;TRAP ON THIS INSTRUCTION
RETR:   CMP     #.,BUFF-4   ;CHECK FOR INCREMENTED P.C.
        BEQ     TST55
        MOV     #216,$FATAL ;MOVE TO MAILBOX # ***** 216 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;INCORRECT P.C.,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 757
  
```

```

010132 005237 000304
010136 022737 000054 000304
010144 001013
1124 010146 012706 000500
1125 010152 012737 010164 000004
1126 010160 005737 000001
1127 010164 022737 010164 000474
1128 010172 001405
    010174 012737 000216 000302
    010202 005212
    010204 000000
  
```

1129

```

.SBTTL TEST #55 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
:TEST 55 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK
:*****
TST55:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #55,$TESTN ;SEQUENCE ERROR?
        BNE     TST56-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP    ;SET UP
        MOV     #RETS,RTRAP5 ;SET UP
        CLR     CC         ;CLEAR CC AND PRIORITY
        CCC
        TST 1
        RETS:  CMP     BUFF-2,#0 ;TRAP
                BEQ     1$      ;TEST THAT OLD STATUS WENT TO STACK
        MOV     #217,$FATAL ;MOVE TO MAILBOX # ***** 217 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;INCORRECT STATUS
                            ;TO SCOPE REPLACE HALT WITH 240
                            ;AND REPLACE NEXT INST WITH 754
1137:  MOV     #BUFF,SP    ;SET UP
1138:  MOV     #RETT,RTRAP5 ;SET UP
1139:  MOV     #357,CC     ;SET PRIORITY
1140:  SCC
1141:  TST 1              ;SET CC
1142:  RETT:  CMP     BUFF-2,#357 ;TRAP
                BEQ     TST56    ;COMPARES STATUS ON STACK
1143:  MOV     #220,$FATAL ;MOVE TO MAILBOX # ***** 220 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;INCORRECT STATUS ON STACK,OR WRONG $STNM
                            ;TO SCOPE REPLACE HALT WITH 240
                            ;AND REPLACE NEXT INST WITH 730
    
```

```

010206 005237 000304
010212 022737 000055 000304
010220 001042
1130 010222 012706 000500
1131 010226 012737 010246 000004
1132 010234 005037 177776
1133 010240 000257
1134 010242 005737 000001
1135 010246 023727 000476 000000
1136 010254 001405
        010256 012737 000217 000302
        010264 005212
        010266 000000

1137 010270 012706 000500 1$:
1138 010274 012737 010316 000004
1139 010302 012737 000357 177776
1140 010310 000277
1141 010312 005737 000001
1142 010316 023727 000476 000357 RETT:
1143 010324 001405
        010326 012737 000220 000302
        010334 005212
        010336 000000
    
```

TEST #56 - TEST THAT 'NEW' STATUS IS CORRECT
1144

.SBTTL TEST #56 - TEST THAT 'NEW' STATUS IS CORRECT

:*****
:TEST 56 - TEST THAT 'NEW' STATUS IS CORRECT
:*****

	010340	005237	000304		TST56:	INC	\$TESTN		;UPDATE TEST NUMBER
	010344	022737	000056	000304		CMP	#56,\$TESTN		;SEQUENCE ERROR?
	010352	001121				BNE	TST57-12		;BR TO ERROR HALT ON SEQ ERROR
1145	010354	012706	000500			MOV	#BUFF,SP		
1146	010360	012737	010376	000004		MOV	#RETU,RTRAP5		
1147	010366	005037	000006			CLR	RTRAP5+2		;CLEAR FUTURE PRICRITY AND CC
1148	010372	005737	000001			TST	1		;TRAP HERE
1149	010376				RETU:				;TEST FOR 'C' CLEARED
1150	010376	100005				BPL	1\$		
	010400	012737	000221	000302		MOV	#221,\$FATAL		;MOVE TO MAILBOX # ***** 221 *****
	010406	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010410	000000				HALT			;C NOT CLEARED
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 760
1151	010412				1\$:				
	010412	001005				BNE	2\$		
	010414	012737	000222	000302		MOV	#222,\$FATAL		;MOVE TO MAILBOX # ***** 222 *****
	010422	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010424	000000				HALT			;Z NOT CLEARED
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 752
1152	010426				2\$:				
	010426	102005				BVC	3\$		
	010430	012737	000223	000302		MOV	#223,\$FATAL		;MOVE TO MAILBOX # ***** 223 *****
	010436	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010440	000000				HALT			;V NOT CLEARED
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 744
1153	010442				3\$:				
	010442	103005				BCC	4\$		
	010444	012737	000224	000302		MOV	#224,\$FATAL		;MOVE TO MAILBOX # ***** 224 *****
	010452	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010454	000000				HALT			;C NOT CLEARED
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 736
1154	010456	032737	000357	177776	4\$:	BIT	#357,CC		;TEST PRIORITY FOR ZERO
1155	010464	001405				BEQ	5\$		
	010466	012737	000225	000302		MOV	#225,\$FATAL		;MOVE TO MAILBOX # ***** 225 *****
	010474	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010476	000000				HALT			;PRIORITY NOT ZERO
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 725
1156	010500	012706	000500		5\$:	MOV	#BUFF,SP		
1157	010504	012737	010524	000004		MOV	#RETV,RTRAP5		
1158	010512	012737	000357	000006		MOV	#357,RTRAP5+2		;SET NEW 'CC' AND PRIORITY
1159	010520	005737	000001			TST	1		;TRACE HERE
1160	010524				RETV:				
1161	010524	100405				BMI	1\$		
	010526	012737	000226	000302		MOV	#226,\$FATAL		;MOVE TO MAILBOX # ***** 226 *****
	010534	005212				INC	(R2)		;SET MSGTYP TO FATAL ERROR
	010536	000000				HALT			;N NOT SET
									;TO SCOPE REPLACE HALT WITH 240
									;AND REPLACE NEXT INST WITH 705

1162	010540			1\$:	BEQ	2\$		
	010540	001405			MOV	#227,\$FATAL	;MOVE TO MAILBOX # ***** 227 *****	
	010542	012737	000227 000302		INC	(R2)	;SET MSGTYP TO FATAL ERROR	
	010550	005212			HALT		;Z NOT SET	
	010552	000000					;TO SCOPE REPLACE HALT WITH 240	
							;AND REPLACE NEXT INST WITH 677	
1163	010554			2\$:	BVS	3\$		
	010554	102405			MOV	#230,\$FATAL	;MOVE TO MAILBOX # ***** 230 *****	
	010556	012737	000230 000302		INC	(R2)	;SET MSGTYP TO FATAL ERROR	
	010564	005212			HALT		;V NOT SET	
	010566	000000					;TO SCOPE REPLACE HALT WITH 240	
							;AND REPLACE NEXT INST WITH 671	
1164	010570			3\$:	BCS	4\$		
	010570	103405			MOV	#231,\$FATAL	;MOVE TO MAILBOX # ***** 231 *****	
	010572	012737	000231 000302		INC	(R2)	;SET MSGTYP TO FATAL ERROR	
	010600	005212			HALT		;C NOT SET	
	010602	000000					;TO SCOPE REPLACE HALT WITH 240	
							;AND REPLACE NEXT INST WITH 663	
1165	010604	013700	177776	4\$:	MOV	CC,%0		
1166	010610	022700	000357		CMP	#357,%0		
1167	010614	001405			BEQ	TST57		
	010616	012737	000232 000302		MOV	#232,\$FATAL	;MOVE TO MAILBOX # ***** 232 *****	
	010624	005212			INC	(R2)	;SET MSGTYP TO FATAL ERROR	
	010626	000000			HALT		;PRIORITY WAS CHANGED,OR WRONG \$STINM	
							;TO SCOPE REPLACE HALT WITH 240	
							;AND REPLACE NEXT INST WITH 651	

1168

.SBTTL TEST #57 - TEST THAT DEC R6 TO A VALUE LESS 400 TRAPS
 :*****
 :TEST 57 - TEST THAT DEC R6 TO A VALUE LESS 400 TRAPS
 :*****

010630 005237 000304
 010634 022737 000057 000304
 010642 001027
 1169 010644 005037 177766
 1170 010650 012706 000150
 1171 010654 012737 010676 000004
 1172 010662 005746
 1173 010664 012737 000233 000302
 010672 005212
 010674 000000

TST57: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #57,\$TESTN ;SEQUENCE ERROR?
 BNE ERRP2 ;BR TO ERROR HALT ON SEQ ERROR
 CLR CPUERR ;CLEAR CPU ERROR REGISTER
 MOV #150,%R6 ;R6 = 150
 MOV #TDEC1,4 ;STACK OVERFLOW TRAP POINTER
 TST -(6) ;WITH R6 = 150 SHOULD TRAP
 MOV #233,\$FATAL ;MOVE TO MAILBOX # ***** 233 *****
 !NC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;SHOULD HAVE TRAPPED,OR WRONG \$STNM
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 762

1174 010676
 1175 010676 013737 177766 000502
 1176 010704 042737 177413 000502
 1177 010712 022737 000004 000502
 1178 010720 001405
 010722
 010722 012737 000234 000302
 010730 005212
 010732 000000

TDEC1: MOV CPUERR,RCPUER ;SAVE CPU ERROR REGISTER
 BIC #CERMSK,RCPUER ;MASK OFF UNUSED CPU ERROR REG BITS
 CMP #4,RCPUER ;IS YELLOW ZONE BIT SET?
 BEQ CERR2
 ERRP2: MOV #234,\$FATAL ;MOVE TO MAILBOX # ***** 234 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERROR REGISTER CONTENTS, OR WRONG \$STNM
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 743
 CERR2: CLR CPUERR ;CLEAR YELLOW ZONE BIT

1179 010734 005037 177766
 1180

1181

```

.SBTTL TEST #60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
:*****
;TEST 60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP
:*****
TST60:  INC      $TESTM      ;UPDATE TEST NUMBER
        CMP      #60,$TESTM ;SEQUENCE ERROR?
        BNE     TST61-12    ;BR TO ERROR HALT ON SEQ ERROR
1182 010740 005237 000304 000304  MOV     #150,%6          ;R6 = 150
1183 010754 012706 000150 000004  MOV     #TDEC2,4        ;TRAP POINTER
1184 010760 012737 010770 000004  TST     -(6)            ;WITH R6 = 150 SHOULD TRAP
1185 010766 005746 000142 000004  TDEC2: CMP     %6,#142   ;DID R6 DECREMENT
1186 010774 001405 000142 000004  BEQ     TST61
        MOV     #235,$FATAL ;MOVE TO MAILBOX # ***** 235 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT                ;R6 NOT = 142,OR WRONG $TSTNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 76;
    
```

1187

1189

.SBTTL TEST #61 - TEST DIFFERENT TYPES OF OVERFLOW
 :*****
 :TEST 61 - TEST DIFFERENT TYPES OF OVERFLOW
 :*****

011010	005237	000304		TST61:	INC	\$TESTN		:UPDATE TEST NUMBER
011014	022737	000061	000304		CMP	#61,\$TESTN		:SEQUENCE ERROR?
011022	001043				BNE	TST62-12		:BR TO ERROR HALT ON SEQ ERROR
1190 011024	012706	000150			MOV	#150,%6		
1191 011030	005037	000146			CLR	146		:STATUS WORD OF LOC 10
1192 011034	012737	011044	000004		MOV	#TDEC3,4		:RETURN TO LOC 4
1193 011042	005246				INC	-(6)		
1194 011044	005737	000146		TDEC3:	TST	146		
1195 011050	001005				BNE	1\$		
	011052	012737	000236	000302	MOV	#236,\$FATAL		:MOVE TO MAILBOX # ***** 236 *****
	011060	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	011062	0000C0			HALT			:INCREMENT OPERATION NOT INHIBITED
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 757
1196 011064	012705	001000		1\$:	MOV	#1000,%5		
1197 011070	012706	000400			MOV	#400,%6		
1198 011074	012737	011116	000004		MOV	#TDEC4,4		
1199 011102	124645				CMPB	-(6),-(5)		
1200 011104	012737	000237	000302		MOV	#237,\$FATAL		:MOVE TO MAILBOX # ***** 237 *****
	011112	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	011114	000000			HALT			:STACK = 400 AND DECREMENTED, SHOULD TRAP
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 742
1201 011116	012706	000400		TDEC4:	MOV	#400,%6		
1202 011122	012737	011144	000004		MOV	#TDEC7,4		
1203 011130	134546				BITB	-(5),-(6)		
1204 011132				TDEC6:				
	011132	012737	000240	000302	MOV	#240,\$FATAL		:MOVE TO MAILBOX # ***** 240 *****
	011140	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	011142	000000			HALT			:NO STACK OVERFLOW,OR WRONG \$STNM
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 727
1205 011144				TDEC7:				
1215								

1216

```

.SBTTL TEST #62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
:*****
:TEST 62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP
:*****
011144 005237 000304 TST62: INC $TESTN ;UPDATE TEST NUMBER
011150 022737 000062 0C0304 CMP #62,$TESTN ;SEQUENCE ERROR?
011156 001011 BNE VDEC2 ;BR TO ERROR HALT ON SEQ ERROR
011160 012706 000400 MOV #400,%6 ;SET UP STACK TO OVERFLOW
011164 012737 011202 0C0010 MOV #VDEC2,10 ;SET UP 10 VECTOR
011172 012737 011214 0C0004 MOV #VDEC1,4 ;SET UP OVERFLOW VECTOR
011200 000010 10 ;THIS TRAP SHOULD CAUSE OVERFLOW
011202 VDEC2:
011202 012737 000241 000302 MOV #241,$FATAL ;MOVE TO MAILBOX # ***** 241 *****
011210 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
011212 000000 HALT ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 761

011214 012737 000012 000010 VDEC1: MOV #10+2,10
  
```

1217

```

.SBTTL TEST #63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
:*****
:TEST 63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP
:*****
TST63:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #63,$TESTN ;SEQUENCE ERROR?
        BNE     VDEC4      ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #400,%6     ;SET UP STACK TO OVERFLOW
        MOV     #VDEC4,20   ;SET UP IOT VECTOR
        MOV     #VDEC3,4    ;SET UP OVERFLOW VECTOR
        IOT     ;THIS TRAP SHOULD CAUSE OVERFLOW

VDEC4:  MOV     #242,$FATAL  ;MOVE TO MAILBOX # ***** 242 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761

011272 012737 000022 000020 VDEC3:  MOV     #20+2,20
  
```

```

011222 005237 000304
011226 022737 000063 000304
011234 001011
011236 012706 000400
011242 012737 011260 000020
011250 012737 011272 000004
011256 000004
011260
011260 012737 000242 000302
011266 005212
011270 000000
  
```

1218

```

.SBTTL TEST #64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
:*****
:TEST 64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
:*****
011300 005237 000304      TST64: INC      $TESTN      ;UPDATE TEST NUMBER
011304 022737 000064 000304    CMP      #54,$TESTN    ;SEQUENCE ERROR?
011312 001011                BNE      VDEC6         ;BR TO ERROR HALT ON SEQ ERROR
011314 012706 000400                MOV      #400,%6       ;SET UP STACK TO OVERFLOW
011320 012737 011336 000030    MOV      #VDEC6,30     ;SET UP EMT VECTOR
011326 012737 011350 000004    MOV      #VDEC5,4      ;SET UP OVERFLOW VECTOR
011334 104000                EMT                    ;THIS TRAP SHOULD CAUSE OVERFLOW
011336                VDEC6: MOV      #243,$FATAL    ;MOVE TO MAILBOX # ***** 243 *****
011336 012737 000243 000302    INC      (R2)          ;SET MSGTYP TO FATAL ERROR
011344 005212                HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TSTNM
011346 000000                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 761
011350 012737 000032 000030    VDEC5: MOV      #30+2,30
  
```

1219

```

.SBTTL TEST #65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
:*****
:TEST 65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
:*****
TST65:  INC    $TESTN      ;UPDATE TEST NUMBER
        CMP    #65,$TESTN ;SEQUENCE ERROR?
        BNE   VDEC8      ;BR TO ERROR HALT ON SEQ ERROR
        MOV   #400,%6     ;SET UP STACK TO OVERFLOW
        MOV   #VDEC8,34   ;SET UP TRAP VECTOR
        MOV   #VDEC7,4    ;SET UP OVERFLOW VECTOR
        TRAP ;THIS TRAP SHOULD CAUSE OVERFLOW

VDEC8:  MOV   #244,$FATAL  ;MOVE TO MAILBOX # ***** 244 *****
        INC   (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TESTN
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761

VDEC7:  MOV   #34+2,34
    
```

```

011356 005237 000304
011362 022737 000065 000304
011370 001011
011372 012706 000400
011376 012737 011414 000034
011404 012737 011426 000004
011412 104400
011414
011414 012737 000244 000302
011422 005212
011424 000000

011426 012737 000036 000034
    
```

1220

```

.SBTTL TEST #66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
:*****
:TEST 66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP
:*****
TST66: INC      $TESTN      ;UPDATE TEST NUMBER
      CMP      #66,$TESTN  ;SEQUENCE ERROR?
      BNE      VDEC10     ;BR TO ERROR HALT ON SEQ ERROR
      MOV      #400,%6     ;SET UP STACK TO OVERFLOW
      MOV      #VDEC10,14  ;SET UP TRT VECTOR
      MOV      #VDEC9,4    ;SET UP OVERFLOW VECTOR
      TRT                      ;THIS TRAP SHOULD CAUSE OVERFLOW

VDEC10: MOV      #245,$FATAL ;MOVE TO MAILBOX # ***** 245 *****
      INC      (R2)        ;SET MSGTYP TO FATAL ERROR
      HALT                    ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TESTN
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 761

VDEC9: MOV      #14+2,14
    
```

```

011434 005237 000304
011440 022737 000066 000304
011446 001011
011450 012706 000400
011454 012737 011472 000014
011462 012737 011504 000004
011470 000003
011472
011472 012737 000245 000302
011500 005212
011502 000000

011504 012737 000016 000014
    
```

1221

```
.SBTTL TEST #67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
:*****
:TEST 67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP
:*****
TST67: INC $TESTN ;UPDATE TEST NUMBER
011512 005237 000304 000304 CMP #67,$TESTN ;SEQUENCE ERROR?
011516 022737 000067 000304 BNE VDEC11 ;BR TO ERROR HALT ON SEQ ERROR
011524 001011 000400 000010 MOV #400,%6 ;SET UP STACK TO OVERFLOW
011526 012706 011550 000010 MOV #VDEC11,10 ;SET UP ILLA VECTOR
011532 012737 011562 000004 MOV #VDEC12,4 ;SET UP OVERFLOW VECTOR
011540 012737 011562 000004 ILLA ;THIS TRAP SHOULD CAUSE OVERFLOW
011546 004700
011550
011550 012737 000246 000302 VDEC11: MOV #246,$FATAL ;MOVE TO MAILBOX # ***** 246 *****
011556 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
011560 000000 HALT ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $TESTN
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 761

1222 011562 012737 000012 000010 VDEC12: MOV #10+2,10
1223 011570 020627 000370 CMP %6,#370 ;STACK PUSHED FOUR WORDS?
011574 001405 BEQ TST70
011576 012737 000247 000302 MOV #247,$FATAL ;MOVE TO MAILBOX # ***** 247 *****
011604 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
011606 000000 HALT ;CORRECT # (4) OF WORDS WERE NOT PUSHED ONTO STACK
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 746
```

1224

```

.SBTTL TEST #70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
;*****
;TEST 70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
;*****
TST70: INC $TESTN ;UPDATE TEST NUMBER
        CMP #70,$TESTN ;SEQUENCE ERROR?
        BNE VDEC13 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #400,%6 ;SET UP STACK TO OVERFLOW
        MOV #VDEC13,10 ;SET UP ILLB VECTOR
        MOV #VDEC14,4 ;SET UP OVERFLOW VECTOR
        ILLB ;THIS TRAP SHOULD CAUSE OVERFLOW

VDEC13: MOV #250,$FATAL ;MOVE TO MAILBOX # ***** 250 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $STNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 761
    
```

C11610 005237 000304
 011614 022737 000070 000304
 011622 001011
 011624 012706 000400
 011630 012737 011646 000010
 011636 012737 011660 000004
 011644 000100
 011646
 011646 012737 000250 000302
 011654 005212
 011656 000000

1225

011660 012737 000012 000010 VDEC14: MOV #10+2,10

1226

.SBTTL TEST #71 - TEST FOR FALSE OVERFLOW TRAP
 :*****
 :TEST 71 - TEST FOR FALSE OVERFLOW TRAP
 :*****

011666	005237	000304		TST71:	INC	\$TESTN	;UPDATE TEST NUMBER	
011672	022737	000071	000304		CMP	#71,\$TESTN	;SEQUENCE ERROR?	
011700	001023				BNE	FOVER	;BR TO ERROR HALT ON SEQ ERROR	
1227								
1228	011702	012737	011750	000004	MOV	#FOVER,4	;SET UP OVERFLOW POINTER	
1229	011710	012706	001002		MOV	#1002,%6		
1230	011714	005746			TST	-(6)	;SHOULD NOT OVERFLOW	
1231	011716	012706	002002		MOV	#2002,%6		
1232	011722	005746			TST	-(6)	;SHOULD NOT OVERFLOW	
1233	011724	012706	004002		MOV	#4002,%6		
1234	011730	005746			TST	-(6)	;SHOULD NOT OVERFLOW	
1235	011732	012706	010002		MOV	#10002,%6		
1236	011736	005746			TST	-(6)		
1237	011740	012706	020000		MOV	#20000,%6	;SHOULD NOT OVERFLOW	
1238	011744	005746			TST	-(6)		
1239	011746	000405			BR	STP		
	011750							
	011750	012737	000251	000302	FOVER:	MOV	#251,\$FATAL	;MOVE TO MAILBOX # ***** 251 *****
	011756	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
	011760	000000				HALT		;IT OVERFLOWED,OR WRONG \$STNM
								;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 747
1240	011762	012737	000006	000004	STP:	MOV	#6,4	
1241	011770	005037	000006			CLR	6	

1242

```

.SBTTL TEST #72 - TEST THAT BIT 4 PSW WILL CAUSE A TRAP TO 14
:*****
:TEST 72 - TEST THAT BIT 4 PSW WILL CAUSE A TRAP TO 14
:*****
TST72:  INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #72,$TESTN  ;SEQUENCE ERROR?
        BNE     TST73-12    ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,SP
        MOV     #RETAT,RTRAP4 ;SET UP TO TRAP TO 14
        MOV     #20,-(SP)    ;PUSH T BIT
        MOV     #.+6,-(SP)   ;PUSH PC
        RTI
        NOP
        MOV     #252,$FATAL  ;MOVE TO MAILBOX # ***** 252 *****
        INC     (R2)         ;SET MSGTYP TO FATAL ERROR
        HALT                ;TRACE BIT DID NOT TRAP!,OR WRONG $TESTN
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 757
    
```

```

011774 005237 000304
012000 022737 000072 000304
012006 001013
1243 012010 012706 000500
1244 012014 012737 012050 000014
1245 012022 012746 000020
1246 012026 012746 012034
1247 012032 000002
1248 012034 000240
1249 012036 012737 000252 000302
      012044 005212
      012046 000000
    
```

1250 012050

RETAT:

1261

.SBTTL TEST #74 - TEST FOR PROPER PC ON STACK
 :*****
 :TEST 74 - TEST FOR PROPER PC ON STACK
 :*****

012144	005237	000304		TST74:	INC	\$TESTN	:UPDATE TEST NUMBER
012150	022737	000074	000304		CMP	#74,\$TESTN	:SEQUENCE ERROR?
012156	001016				BNE	TST75-12	:BR TO ERROR HALT ON SEQ ERROR
1262	012160	012706	000500		MOV	#BUFF,SP	
1263	012164	012737	012204	000014	MOV	#RETCT,RTRAP4	
1264	012172	012746	000020		MOV	#20,-(SP)	:PUSH T BIT
1265	012176	012746	012204		MOV	#.+6,-(SP)	:PUSH PC
1266	012202	000002			RTI		:SET T BIT
1267							:TRAP HERE
1268	012204	022737	012204	000474	RETCT:	CMP	#.,BUFF-4
1269	012212	001405			BEQ	TST75	
	012214	012737	000255	000302	MOV	#255,\$FATAL	:MOVE TO MAILBOX # ***** 255 *****
	012222	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	012224	000000			HALT		:CORRECT PC WAS NOT SAVED ON STACK,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 754

1270
1271

1272

.SBTTL TEST #75 - TEST THAT RTT POPS T- BIT
 :*****
 :TEST 75 - TEST THAT RTT POPS T- BIT
 :*****

012226 005237 000304
 012232 022737 000075 000304
 012240 001015

TST75: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #75,\$TESTN ;SEQUENCE ERROR?
 BNE TST76-12 ;BR TO ERROR HALT ON SEQ ERROR

1273

1274 012242 012706 000500
 1275 012246 005001
 1276 012250 012746 000020
 1277 012254 012746 012270
 1278 012260 012737 012306 000014
 1279 012266 000006
 1280 012270 000240
 1281 012272 001405
 012274 012737 000256 000302
 012302 005212
 012304 000000

MOV #BUFF,SP
 CLR R1 ;CLEAR R1
 MOV #20,-(SP)
 MOV #RTT1,-(SP)
 MOV #RTT2,14
 RTT
 RTT1: NOP
 BEQ TST76
 MOV #256,\$FATAL ;MOVE TO MAILBOX # ***** 256 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;T-BIT DID NOT TRAP,OR WRONG \$TESTN
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 755

1282

1283 012306

RTT2:

1284

```

.SBTTL TEST #76 - TEST THAT RTT ALLOWS ONE INST. BEFORE TRAP
:*****
:TEST 76 - TEST THAT RTT ALLOWS ONE INST. BEFORE TRAP
:*****
TST76:  INC  $TESTN      ;UPDATE TEST NUMBER
        CMP  #76,$TESTN ;SEQUENCE ERROR?
        BNE  TST77-12   ;BR TO ERROR HALT ON SEQ ERROR
        MOV  #177777,%5
RTT5:   MOV  #BUFF,SP
        MOV  #20,-(SP)
        MOV  #RTT3,-(SP)
        MOV  #RTT4,14
        CLR  R1          ;CLEAR R0
        RTT          ;SET T-BIT
RTT3:   INC  R1
        INC  %5
        BEQ  RTT5        ;DO THIS TEST NO MORE THAN 2 TIMES
        MOV  #257,$FATAL ;MOVE TO MAILBOX # ***** 257 *****
        INC  (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT            ;DID NOT TRAP
                        ;TO SCOPE REPLACE HALT WITH 240
                        ;AND REPLACE NEXT INST WITH 752
                        ;SEE IF RTT ALLOWS 1 INST.
RTT4:   DEC  R1
        BEQ  RTT6
        INC  %5          ;DO THIS TEST NO MORE THAN TWO TIMES
        BEQ  RTT5
        MOV  #260,$FATAL ;MOVE TO MAILBOX # ***** 260 *****
        INC  (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT            ;RTT DID NOT ALLOW 1 INST.,OR WRONG $TESTN
                        ;TO SCOPE REPLACE HALT WITH 240
                        ;AND REPLACE NEXT INST WITH 741
RTT6:
    
```

```

012306 005237 000304
012312 022737 000076 000304
012320 001031
1285 012322 012705 177777
1286 012326 012706 000500
1287 012332 012746 000020
1288 012336 012746 012354
1289 012342 012737 012374 000014
1290 012350 005001
1291 012352 000006
1292 012354 005201
1293 012356 005205
1294 012360 001762
1295 012362 012737 000257 000302
    012370 005212
    012372 000000

1296 012374 005301
1297 012376 001407
1298 012400 005205
1299 012402 001751
    012404 012737 000260 000302
    012412 005212
    012414 000000

1300 012416
    
```

1301

.SBTTL TEST #77 - TEST THAT RTI DOES NOT ALLOW 1 INST.
 ;*****
 ;TEST 77 - TEST THAT RTI DOES NOT ALLOW 1 INST.
 ;*****

012416 005237 000304
 012422 022737 000077 000304
 012430 001023
 1302 012432 012706 000500
 1303 012436 012746 000020
 1304 012442 012746 012460
 1305 012446 012737 012474 000014
 1306 012454 005001
 1307 012456 000002
 1308 012460 005201
 1309 012462 012737 000261 000302
 012470 005212
 012472 000000

TST77: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #77,\$TESTN ;SEQUENCE ERROR?
 BNE TST100-12 ;BR TO ERROR HALT ON SEQ ERROR
 MOV #BUFF,SP
 MOV #20,-(SP)
 MOV #RTI1,-(SP)
 MOV #RTI2,14
 CLR R1
 RTI ;SET T-BIT
 RTI1: INC R1 ;RTI SHOULD NOT ALLOW THIS
 MOV #261,\$FATAL ;MOVE TO MAILBOX # ***** 261 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;T- BIT DID NOT CAUSE TRAP
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 756
 RTI2: TST R1
 ;RTI SHOULD NOT ALLOW 1 INST. BEFORE TRAP
 BEQ TST100
 MOV #262,\$FATAL ;MOVE TO MAILBOX # ***** 262 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;RTI DID ALLOW 1 INST. BEFORE TRAP,OR WRONG \$TESTN
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 747

1310 012474 005701
 1311
 1312 012476 001405
 012500 012737 000262 000302
 012506 005212
 012510 000000

1313

J 7
 .SBTTL TEST #100 - DOES THE PROCESSOR TRAP WHEN %7 IS ODD?
 :*****
 :TEST 100 - DOES THE PROCESSOR TRAP WHEN %7 IS ODD?
 :*****

```

012512 005237 000304
012516 022737 000100 000304
012524 001120
1315 012526 012706 000500
1316 012532 012737 012556 000004
1317 012540 012707 000001
1318 012544 012737 000263 000302
012552 005212
012554 000000

TST100: INC $TESTN ;UPDATE TEST NUMBER
        CMP #100,$TESTN ;SEQUENCE ERROR?
        BNE TST101-12 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,%6 ;SET UP STACK POINTER
        MOV #R7TR1,4 ;RETURN FROM TRAP
        MOV #1,%7 ;PC EQUALS ONE
        MOV #263,$FATAL ;MOVE TO MAILBOX # ***** 263 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;ODD ADDRESS SHOULD HAVE TRAPPED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 763

1319 012556 022737 000001 000474 R7TR1: CMP #1,BUFF-4
1320 012564 001405 BEQ 1$
012566 012737 000264 000302 MOV #264,$FATAL ;MOVE TO MAILBOX # ***** 264 *****
012574 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
012576 000000 HALT ;CORRECT PC WAS NOT SAVED ON STACK
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 752

1321
1322 012600 012706 000500 1$: MOV #BUFF,%6 ;STACK POINTER
1323 012604 012737 012626 000004 MOV #R7TR2,4
1324 012612 005207 INC %7 ;PC BECOMES ODD
1325 012614 R7TR2A: MOV #265,$FATAL ;MOVE TO MAILBOX # ***** 265 *****
012614 012737 000265 000302 INC (R2) ;SET MSGTYP TO FATAL ERROR
012622 005212 HALT ;CORRECT PC WAS NOT SAVED ON STACK
012624 000000 ;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 737

1326 012626 022737 012615 000474 R7TR2: CMP #R7TR2A+1,BUFF-4
1327 012634 001405 BEQ 1$
012636 012737 000266 000302 MOV #266,$FATAL ;MOVE TO MAILBOX # ***** 266 *****
012644 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
012646 000000 HALT ;CORRECT PC NOT ON STACK
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 726

1328 012650 012706 000500 1$: MOV #BUFF,%6
1329 012654 012737 012676 000004 MOV #R7TR3,4
1330 012662 005307 BR60: DEC %7 ;MAKE PC ODD
1331 012664 012737 000267 000302 MOV #267,$FATAL ;MOVE TO MAILBOX # ***** 267 *****
012672 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
012674 000000 HALT ;SHOULD TRAP
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 713

1332 012676 022737 012663 000474 R7TR3: CMP #BR60+1,BUFF-4
1333 012704 001405 BEQ 1$
012706 012737 000270 000302 MOV #270,$FATAL ;MOVE TO MAILBOX # ***** 270 *****
012714 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
012716 000000 HALT ;WRONG VALUE ON STACK
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 702

1334
1335 012720 012706 000500 1$: MOV #BUFF,%6
1336 012724 012737 012750 000004 MOV #R7TR4,4
    
```

```

1337 012732 000261          SEC          ;CARRY EQUALS A 1
1338 012734 006107          ROL          %7          ;PC BECOMES ODD
1339 012736          TR4A:  MOV          #271,$FATAL      ;MOVE TO MAILBOX # ***** 271 *****
      012736 012737 000271 000302  INC          (R2)          ;SET MSGTYP TO FATAL ERROR
      012744 005212          HALT          ;ODD ADDRESS DIDN'T TRAP
      012746 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 666
      ;RESET UP A HALT FOR TRAP

1340 012750 012737 000006 000004  R7TR4:  MOV          #6,4
1341 012756 022737 025675 000474  CMP          #<2*TR4A+1> ,BUFF-4 ;CHECK FOR VALUE ON STACK
1342 012764 001405          BEQ          TST101
      012766 012737 000272 000302  MOV          #272,$FATAL      ;MOVE TO MAILBOX # ***** 272 *****
      012774 005212          INC          (R2)          ;SET MSGTYP TO FATAL ERROR
      012776 000000          HALT          ;WRONG VALUE ON STACK,OR WRONG $STNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 652
  
```


1343

.SBTTL TEST #101 - TEST THAT TRACE BIT TRAPS INHIB ON TRAP INST
 :*****
 :TEST 101 - TEST THAT TRACE BIT TRAPS INHIB ON TRAP INST
 :*****

013000	005237	000304		TS1101: INC	\$TESTN	:UPDATE TEST NUMBER
013004	022737	000101	000304	CMP	#101,\$TESTN	:SEQUENCE ERROR?
013012	001027			BNE	BR70	:BR TO ERROR HALT ON SEQ ERROR
1344						
1345	013014	012706	000500	MOV	#BUFF,%6	
1346	013020	012737	013060	MOV	#TRACE,14	:TRACE TRAP
1347	013026	005027	000016	CLR	#16	
1348	013032	005027	000022	CLR	#22	
1349	013036	012737	013104	MOV	#TONT1,20	:IOT TRAP
1350	013044	012746	000020	MOV	#20,-(SP)	:PUSH T BIT
1351	013050	012746	013056	MOV	#+6,-(SP)	:PUSH PC
1352	013054	000006		RTT		
1353	013056	000004		IOT		:TRAP, NEW CC HAVE TRACE RESET
1354	013060			TRACE:		
	013060	012737	000273	MOV	#273,\$FATAL	:MOVE TO MAILBOX # ***** 273 *****
	013066	005212		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	013070	000000		HALT		:TRACE TRAP WAS NOT INHIBITED
						:TO SCOPE REPLACE HALT WITH 240
						:AND REPLACE NEXT INST WITH 750
1355	013072			BR70:		
	013072	012737	000274	MOV	#274,\$FATAL	:MOVE TO MAILBOX # ***** 274 *****
	013100	005212		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	013102	000000		HALT		:WRONG TSTNM,OR WRONG \$TSTNM
						:TO SCOPE REPLACE HALT WITH 240
						:AND REPLACE NEXT INST WITH 743
1356	013104	012737	000016	TONT1: MOV	#16,14	
1357	013112	012737	000022	MOV	#22,20	

1358

```

.SBTTL TEST #102 - TEST THAT THE TRACE BIT IS SAVED IN THE STACK
:*****
:TEST 102 - TEST THAT THE TRACE BIT IS SAVED IN THE STACK
:*****
TST102: INC $STSN ;UPDATE TEST NUMBER
        CMP #102,$STSN ;SEQUENCE ERROR?
        BNE STP3 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #BUFF,%6 ;SET UP STACK POINTER
        MOV #TRC1,14 ;TRACE TRAP RETURN
        CLR 16
        MOV #20,-(SP) ;SET THE T BIT
        MOV #TRC1,-(SP)
        RTI
TRC1: BIT BUFF-2,#20 ;CHECK FOR T BIT ON STACK
      BNE STP3D
STP3: MOV #275,$FATAL ;MOVE TO MAILBOX # ***** 275 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;T BIT NOT SAVED ON THE STACK,OR WRONG $STSNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 752
  
```

013120 005237 000304
 013124 022737 000102 000304
 013132 001020
 1359 013134 012706 000500
 1360 013140 012737 013164 000014
 1361 013146 005037 000016
 1362 013152 012746 000020
 1363 013156 012746 013164
 1364 013162 000002
 1365 013164 033727 000476 000020
 1366 013172 001005
 013174
 013174 012737 000275 000302
 013202 005212
 013204 000000

 1367 013206 012737 000016 000014
 1368
 1369
 1370
 1371

.THIS ROUTINE TEST THAT NO LEGAL ADDRESS TRAPS.
 ;AND THAT AN ILLEGAL ADDRESS TRAPS TO LOCATION 4

013214 005237 000304
 013220 022737 000103 000304
 013226 001160

.SBTTL TEST #103 - TEST NON-EXISTENT ADDRESS TRAPS
 :*****
 :TEST 103 - TEST NON-EXISTENT ADDRESS TRAPS
 :*****

TST103: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #103,\$TESTN ;SEQUENCE ERROR?
 BNE AUTO1 ;BR TO ERROR HALT ON SEQ ERROR

1373
 1374
 1375 013230 000402
 1376 013232 000000
 1377 013234 000000
 1378 013236 005000
 1379 013240 005037 177766
 1380 013244 005037 000006
 1381 013250 012737 013304 000004
 1382 013256 012706 000500
 1383 013262 105720
 1384 013264 020027 160000
 1385 013270 101772
 1386 013272
 013272 012737 000276 000302
 013300 005212
 013302 000000

;THIS ROUTINE TESTS MEMORY UNTIL IT DOES A NXM TRAP
 BR ADALL
 TSL: 0
 CORH: 0
 ADALL: CLR %0
 CLR CPUERR ;CLEAR CPU ERROR REGISTER
 CLR 6
 MOV #ATRAP,4 ;SET UP ADDRESS TRAP ENTRANCE
 NOR: MOV #BUFF,SP
 TSTB (0)+ ;IF OUTSIDE OF CORE, TRAP TO 4
 CMP %0,#160000 ;IS POINTER IN SIDE CORE
 BLOS NOR ;TEST THE REST OF CORE
 AUTO: MOV #276,\$FATAL ;MOVE TO MAILBOX # ***** 276 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;SHOULD HAVE TRAPED
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 751

1387
 1388 013304 010037 013234
 1389 013310 013737 177766 000502
 1390 013316 042737 177413 000502
 1391 013324 020027 160000
 1392
 1393
 1394 013330 103012
 1395 013332 022737 000040 000502
 1396 013340 001417
 013342 012737 000277 000302
 013350 005212
 013352 000000

;RETURN HERE ON AN ADDRESS TRAP
 ATRAP: MOV RO,CORH ;MOVE THE FIRST NXM LOCATION IN CORH
 MOV CPUERR,RCPUER ;SAVE CPU ERROR REGISTER
 BIC #CERMSK,RCPUER ;MASK OFF UNUSED ERROR REG BITS
 CMP %0,#160000 ;WHICH CPU ERROR REG BIT SHOULD BE
 ; SET - NON EXISTANT MEMORY (BIT 5)
 ; OR UNIBUS TIMEOUT (BIT 4)
 BHS 1\$;BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET
 CMP #40,RCPUER ;IS NON-EXISTANT MEMORY BIT SET?
 BEQ 2\$
 MOV #277,\$FATAL ;MOVE TO MAILBOX # ***** 277 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERROR REG CONTENTS
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 725

1397 013354 000411
 1398 013356 022737 000020 000502
 1399 013364 001405
 013366 012737 000300 000302
 013374 005212
 013376 000000

BR 2\$
 1\$: CMP #20,RCPUER ;IS UNIBUS TIMEOUT BIT SET?
 BEQ 2\$
 MOV #300,\$FATAL ;MOVE TO MAILBOX # ***** 300 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERROR REG CONTENTS
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 713

1400
 1401 013400 012700 160001
 1402 013404 005037 177766
 1403 013410 012737 013450 000004
 1404 013416 012706 000500
 1405 013422 105740
 1406 013424 005200
 1407 013426 020037 013234

;THIS ROUTINE DOES NXM TRAPS UNTIL IT FINDS AN EXISTANT MEMORY LOCATION
 2\$: MOV #160001,RO ;SET UP THE HIGHEST MEM LOCATION
 CTRAP: CLR CPUERR ;CLEAR CPU ERROR REGISTER
 MOV #BTRAP,4 ;SET UP THE VECTOR
 MOV #BUFF,SP
 TSTB -(RO) ;DOES IT EXIST?
 DTRAP: INC RO ;IF YES INCREMENT IT
 CMP RO,CORH ;IS IT THE SAME LOCATION?

TEST #103 - TEST NON-EXISTENT ADDRESS TRAPS

```

1408 013432 001463      BEQ      TRAPB
      013434 012737 000301 000302    MOV      #301,$FATAL ;MOVE TO MAILBOX # ***** 301 *****
      013442 005212      INC      (R2)         ;SET MSGTYP TO FATAL ERROR
      013444 000000      HALT                    ;CONTENTS OF R0 AND CORM SHOULD HAVE BEEN EQUAL
                                     ;TO SCOPE REPLACE HALT WITH 240
                                     ;AND REPLACE NEXT INST WITH 670
                                     ;IF THIS COMPARISON FAILS IT MEANS
                                     ;THAT SOME LEGAL ADDRESS TRAPPED OR
                                     ;THAT AN ILLEGAL ADDRESS DID NOT TRAP

1409
1410
1411
1412 013446 000455      BR TRAPB

1413
1414 013450 005737 177776      BTRAP: TST STATUS
1415 013454 001405      BEQ      3$
      013456 012737 000302 000302    MOV      #302,$FATAL ;MOVE TO MAILBOX # ***** 302 *****
      013464 005212      INC      (R2)         ;SET MSGTYP TO FATAL ERROR
      013466 000000      HALT                    ;NEW PSW SHOULD HAVE BEEN ZERO
                                     ;TO SCOPE REPLACE HALT WITH 240
                                     ;AND REPLACE NEXT INST WITH 657

1416
1417 013470 013737 177766 000502 3$:  MOV      CPUERR,RCPUER ;SAVE CPU ERROR REGISTER
1418 013476 042737 177413 000502    BIC      #CERMSK,RCPUER ;MASK OFF UNUSED ERROR REG BITS
1419 013504 020027 160000      CMP      %0,#160000 ;WHICH CPU ERROR REG BIT SHOULD BE
1420
1421
1422 013510 103012      BHIS     1$             ;BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET
1423 013512 022737 000040 000502    CMP      #40,RCPUER ;IS NON-EXISTANT MEMORY BIT SET?
1424 013520 001417      BEQ      2$
      013522 012737 000303 000302    MOV      #303,$FATAL ;MOVE TO MAILBOX # ***** 303 *****
      013530 005212      INC      (R2)         ;SET MSGTYP TO FATAL ERROR
      013532 000000      HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                     ;TO SCOPE REPLACE HALT WITH 240
                                     ;AND REPLACE NEXT INST WITH 635

1425 013534 000411      BR 2$
1426 013536 022737 000020 000502 1$:  CMP      #20,RCPUER ;IS UNIBUS TIMEOUT BIT SET?
1427 013544 001405      BEQ      2$
      013546 012737 000304 000302    MOV      #304,$FATAL ;MOVE TO MAILBOX # ***** 304 *****
      013554 005212      INC      (R2)         ;SET MSGTYP TO FATAL ERROR
      013556 000000      HALT                    ;INCORRECT CPU ERROR REG CONTENTS
                                     ;TO SCOPE REPLACE HALT WITH 240
                                     ;AND REPLACE NEXT INST WITH 623

1428 013560 023727 000474 013424 2$:  CMP      BUFF-4,#DTRAP
1429 013566 001706      BEQ      CTRAP
      013570
      013570 012737 000305 000302    AUTO1: MOV      #305,$FATAL ;MOVE TO MAILBOX # ***** 305 *****
      013576 005212      INC      (R2)         ;SET MSGTYP TO FATAL ERROR
      013600 000000      HALT                    ;OLD PC WAS NOT SAVED OR WRONG $TESTN
                                     ;TO SCOPE REPLACE HALT WITH 240
                                     ;AND REPLACE NEXT INST WITH 612

1430 013602 012737 000006 000004 TRAPB: MOV      #6,4
1431 013610 005037 000006      CLR      6
1432
1433
1434 013614 005037 013640      CLR      PROFTE
1435 013620 012706 000500      MOV      #BUFF,SP ;SET UP THE STACK POINTER
1436 013624 012737 013642 000004    MOV      #DL11W,4 ;SET UP THE TRAP VECTOR
1437 013632 005737 177564      TST     TPS ;TEST THE PUNCH STATUS REGISTER
1438 013636 000403      BR      DL11W1 ;BRANCH IF IT EXISTS

```

1439	013640	000000			PROFTE: 000000		
1440	013642	005237	013640		DL11W: INC	PROFTE	; INCREMENT IF NO DL11W
1441	013646	012737	000006	000004	DL11W1: MOV	#6,4	
1442							

1443

.SBTTL TEST #104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
 :*****
 :TEST 104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
 :*****

013654	005237	000304			TST104: INC	\$STSTN		:UPDATE TEST NUMBER
013660	022737	000104	000304		CMP	#104,\$STSTN		:SEQUENCE ERROR?
013666	001031				BNE	TDEC8		:BR TO ERROR HALT ON SEQ ERROR
1444	013670	005737	013640		TST	PROFTE		
1445	013674	001042			BNE	R7TRX		
1446	013676	000005			RESET			
1447	013700	012737	000340	177776	MOV	#340,\$STATUS		:LOCK OUT INTERRUPT
1448	013706	012706	000400		MOV	#400,%6		:SET UP STACK TO OVERFLOW
1449	013712	012737	013764	000004	MOV	#TDEC77,4		:SET UP OVERFLOW TRAP
1450	013720	012737	013752	000064	MOV	#TDEC8,64		:SET UP INTERRUPT VECTOR
1451	013726	012737	000100	177564	MOV	#100,\$TCSR		:SET INTERRUPT ENABLE
1452	013734	005037	177776		CLR	\$STATUS		:ALLOW INTERRUPT TO OCCUR
1453	013740	012737	000306	000302	MOV	#306,\$FATAL		:MOVE TO MAILBOX # ***** 306 *****
	013746	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	013750	000000			HALT			:NO INTERRUPT OCCURRED
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 746
1454	013752				TDEC8:			
	013752	012737	000307	000302	MOV	#307,\$FATAL		:MOVE TO MAILBOX # ***** 307 *****
	013760	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	013762	000000			HALT			:OVERFLOW TRAP DID NOT OCCUR OR WRONG \$STSTN
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 741
1455	013764	005037	177564		TDEC77:	CLR	\$TCSR	:CLEAR INTERRUPT ENABLE
1456	013770	012737	000006	000004	MOV	#6,4		
1457	013776	005037	000006		CLR	6		
1458	014002				R7TRX:			

1459

.SBTTL TEST #105 - TEST THAT A TRAP OCCURS BEFORE INRUPT
 :*****
 :TEST 105 - TEST THAT A TRAP OCCURS BEFORE INRUPT
 :*****

```

014002 005237 000304
014006 022737 000105 000304
014014 001037
1460 014016 005737 013640
1461 014022 001046
1462 014024 012706 000500
1463 014030 012737 000340 177776
1464 014036 012737 014102 000064
1465 014044 012737 000100 177564
1466 014052 012737 014126 000034
1467 014060 012737 014114 000064
1468 014066 012737 000340 000036
1469 014074 005037 177776
1470 014100 104400
1471 014102
    014102 012737 000310 000302
    014110 005212
    014112 000000

1472 014114
    014114 012737 000311 000302
    014122 005212
    014124 000000

1473 014126 005037 000036
1474 014132 042737 000100 177564
1475 014140
    
```

```

TST105: INC $TESTN ;UPDATE TEST NUMBER
        CMP #105,$TESTN ;SEQUENCE ERROR?
        BNE TR2 ;BR TO ERROR HALT ON SEQ ERROR
        TST PROFTE
        BNE NODL
        MOV #BUFF,%6
        MOV #340,STATUS ;SET TO A HIGH PRIORITY LEVEL
        MOV #TR0,64
        MOV #100,TTCSR ;INTERRUPT FOR TTY PUNCH/PRINTER
        MOV #BR71,34 ;TRAP VECTOR
        MOV #TR2,64 ;TTY VECTOR
        MOV #340,36 ;IF TRAP TRAPS, MOVE 340 TO PRIORITY
        CLR STATUS ;SHOULD INTERRUPT AT END OF CLR INST
        TRAP ;TTY INTERRUPT SHOULD OVERRIDE TRAP

TR0: MOV #310,$FATAL ;MOVE TO MAILBOX # ***** 310 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;NEITHER TRAP NOR INRUPT OCCURED
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 740

TR2: MOV #311,$FATAL ;MOVE TO MAILBOX # ***** 311 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INRUPT OCCURRED FIRST,OR WRONG $TESTNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 733

BR71: CLR 36
      BIC #100,TTCSR

NODL:
    
```

1476

```

.SBTTL TEST #106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
:*****
:TEST 106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
:*****
TST106: INC $TESTN ;UPDATE TEST NUMBER
        CMP #106,$TESTN ;SEQUENCE ERROR?
        BNE TR5 ;BR TO ERROR HALT ON SEQ ERROR
        TST PROFTE
        BNE NODL1
        MOV #BUFF,%6
        MOV #340,STATUS
        MOV #100,TTCSR
        MOV #TR3,34 ;TRAP
        MOV #TR4,64 ;TTY OUTPUT
        MOV #TR5,20 ;IOT
        MOV #340,22 ;IOT PRIORITY
        TRAP ;THE ACT OF TRAPPING LOWER PRIORITY
        TR3: IOT ;INTERRUPT SHOULD OCCUR IN PLACE OF IOT TRAP
        TR5:
        MOV #312,$FATAL ;MOVE TO MAILBOX # ***** 312 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;NO INTERRUPT BETWEEN TRAPS,OR WRONG $STNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 741
        ;CLR IOT PRIORITY
  
```

```

014140 005237 000304
014144 022737 000106 000304
014152 001031
1477 014154 005737 013640
1478 014160 001046
1479 014162 012706 000500
1480 014166 012737 000340 177776
1481 014174 012737 000100 177564
1482 014202 012737 014234 000034
1483 014210 012737 014250 000064
1484 014216 012737 014236 000020
1485 014224 012737 000340 000022
1486 014232 104400
1487 014234 000004
1488 014236
    014236 012737 000312 000302
    014244 005212
    014246 000000

1489 014250 005037 000022
1490 014254 012737 000036 000034
1491 014262 012737 000066 000064
1492 014270 012737 000022 000020
1493 014276
1494
  
```


1495

.SBTTL TEST #107 - TEST THAT 'RESET' GOES TO OUTSIDE WORLD
 :.....
 :TEST 107 - TEST THAT 'RESET' GOES TO OUTSIDE WORLD
 :.....

014276	005237	000304			TST107: INC	\$TESTN		:UPDATE TEST NUMBER
014302	022737	000107	000304		CMP	#107,\$TESTN		:SEQUENCE ERROR?
014310	001027				BNE	TST110-12		:BR TO ERROR HALT ON SEQ ERROR
1496 014312	005737	013640			TST	PROFTE		
1497 014316	001031				BNE	NODL2		
1498 014320	012737	000100	177564		MOV	#100,TTCSR		:SET INTERRUPT ENABLE
1499 014326	012737	000100	177560		MOV	#100,TRCSR		:SET INTERRUPT ENABLE
1500 014334	000005				RESET			:SHOULD CLEAR INTERRUPT ENABLE
1501 014336	032737	000100	177564		BIT	#100,TTCSR		:TEST FOR CLEAR
1502 014344	001405				BEQ	1\$		
	014346	012737	000313	000302	MOV	#313,\$FATAL		:MOVE TO MAILBOX # ***** 313 *****
	014354	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	014356	000000			HALT			:RESET FAILED TO CLEAR TTCSR
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 754
								:TEST FOR CLEAR
1503 014360	032737	000100	177560	1\$:	BIT	#100,TRCSR		
1504 014366	001405				BEQ	TST110		
	014370	012737	000314	000302	MOV	#314,\$FATAL		:MOVE TO MAILBOX # ***** 314 *****
	014376	005212			INC	(R2)		:SET MSGTYP TO FATAL ERROR
	014400	000000			HALT			:RESET FAILED TO CLEAR TRCSR,OR WRONG \$STNM
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 743
1505 014402					NODL2:			

1506

```

.SBTTL TEST #110 - TEST THAT RESET HAS NO EFFECT ON TRACE TRAP
:*****
:TEST 110 - TEST THAT RESET HAS NO EFFECT ON TRACE TRAP
:*****
TST110: INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #110,$TESTN ;SEQUENCE ERROR?
        BNE     RESET3      ;BR TO ERROR HALT ON SEQ ERROR
        MOV     #BUFF,%6     ;SET STACK
        MOV     #RESET2,14   ;SET UP TRACE VECTOR
        MOV     #20,-(R6)    ;SET THE T-BIT ON STACK
        MOV     #1$,-(R6)    ;MOVE NEW PC ON STACK
        RTT
1$:     RESET
        RESET                ;SHOULD HAVE NO EFFECT
                                ;NO EFFECT
RESET3: MOV     #315,$FATAL  ;MOVE TO MAILBOX # ***** 315 *****
        INC     (R2)         ;SET MSGTYP TO FATAL ERROR
        HALT                ;TRACE TRAP FAILED,OR WRONG $STNM
                                ;TO SCOPE REPLACE HALT WITH 240
                                ;AND REPLACE NEXT INST WITH 756
RESET2: CLR     STATUS      ;CLEAR TRACK
        CLR     16          ;TRACE STATUS
        MOV     #16,14
  
```

```

014402 005237 000304
014406 022737 000110 000304
014414 001014
1507 014416 012706 000500
1508 014422 012737 014460 000014
1509 014430 012746 000020
1510 014434 012746 014442
1511 014440 000006
1512 014442 000005
1513 014444 000005
1514 014446
    014446 012737 000315 000302
    014454 005212
    014456 000000

1515 014460 005037 177776
1516 014464 005037 000016
1517 014470 012737 000016 000014
1518
  
```

1519

.SBTTL TEST #111 - TEST THAT WHEN TTY INRUPTS IT POPS NEW STATUS
 :*****
 :TEST 111 - TEST THAT WHEN TTY INRUPTS IT POPS NEW STATUS
 :*****

014476	005237	000304		TST111:	INC	\$TESTN	:UPDATE TEST NUMBER
014502	022737	000111	000304		CMP	#111,\$TESTN	:SEQUENCE ERROR?
014510	001051				BNE	TTY11	:BR TO ERROR HALT ON SEQ ERROR
1520 014512	005737	013640			TST	PROFTE	
1521 014516	001055				BNE	NODL3	
1522 014520	000005				RESET		
1523 014522	012706	000500			MOV	#BUFF,%6	:SET UP STACK
1524 014526	012737	014552	000064		MOV	#TTY3,64	:INTERRUPT VECTOR
1525 014534	005037	177776			CLR	STATUS	:DROP PROCESSOR PRIORITY
1526 014540	012737	000357	000066		MOV	#357,66	:HIGH PRIORITY ON INTERRUPT
1527 014546	005137	177564			COM	TTCSR	:SHOULD SET INTERRUPT ENABLE & INTERRUPT
1528 014552	023727	177776	000357	TTY3:	CMP	STATUS,#357	
1529 014560	001405				BEQ	1\$	
014562	012737	000316	000302		MOV	#316,\$FATAL	:MOVE TO MAILBOX # ***** 316 *****
014570	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014572	000000				HALT		:INTERRUPT DID NOT POP CORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 746
1530 014574	000005			1\$:	RESET		:CLR INTERRUPT ENABLE
1531 014576	012706	000500			MOV	#BUFF,%6	:STACK SET UP
1532 014602	012737	014626	000064		MOV	#TTY4,64	:INTERRUPT VECTOR
1533 014610	005037	000066			CLR	66	:CLR NEW STATUS
1534 014614	012737	000157	177776		MOV	#157,STATUS	:PROCESSOR STATUS
1535 014622	005137	177564			COM	TTCSR	:SET INTERRUPT ENABLE
1536 014626	005737	177776		TTY4:	TST	STATUS	
1537 014632	001405				BEQ	TTT37	
014634				TTY11:			
014634	012737	000317	000302		MOV	#317,\$FATAL	:MOVE TO MAILBOX # ***** 317 *****
014642	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014644	000000				HALT		:INCORRECT STATUS,OR WRONG \$TSTNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 721
1538 014646	005037	177564		TTT37:	CLR	TTCSR	
1539 014652				NODL3:			
1540							

1541

```

.SBTTL TEST #112 - TEST THE 'WAIT' INSTRUCTION
:*****
:TEST 112 - TEST THE 'WAIT' INSTRUCTION
:*****
TST112: INC      $TESTN      ;UPDATE TEST NUMBER
        CMP      #112,$TESTN ;SEQUENCE ERROR?
        BNE      WATE5      ;BR TO ERROR HALT ON SEQ ERROR
        TST     PROFTE
        BNE     NODL4
        BIC     #100,TPS      ;CLEAR INTERRUPT ENABLE
        MOV     #BUFF,SP     ;SET UP THE STACK
        MOV     #WATE,64     ;SET UP THE INTERRUPT VECTOR
        CLR     66
WATE1:  TSTB    TPS          ;WAIT FOR READY
        BPL    WATE1        ;TO BE UP
        MOV     #15,TPB     ;DO A CARRIAGE RETURN
WATE2:  TSTB    TPS          ;WAIT FOR READY TO COME UP
        BPL    WATE2
        MOV     #15,TPB     ;DO ANOTHER CARRIAGE RETURN
        BIS     #100,TPS    ;SET THE INTERRUPT ENABLE
        CLR     STATUS      ;CLEAR THE PSW
WATE3:  WAIT    ;WAIT FOR THE INTERRUPT
        MOV     #320,$FATAL  ;MOVE TO MAILBOX # ***** 320 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT   ;WAIT INSTRUCTION DID NOT LOOP
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 733
WATE:   TST     STATUS ;IS THE PSW CORRECT?
        BEQ    1$
        MOV     #321,$FATAL  ;MOVE TO MAILBOX # ***** 321 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT   ;NEW PSW SHOULD HAVE BEEN ZERO
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 723
1$:     CMP     BUFF-4,#WATE3+2 ;IS THE OLD PC SAVED
        BEQ    WATE4
WATE5:  MOV     #322,$FATAL  ;MOVE TO MAILBOX # ***** 322 *****
        INC     (R2)        ;SET MSGTYP TO FATAL ERROR
        HALT   ;OLD PC WAS NOT SAVED OR WRONG $TESTN
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 712
WATE4:  CLR     TPS          ;CLEAR INTERRUPT ENABLE
NODL4:
    
```

014652 005237 000304
 014656 022737 000112 000304
 014664 001060
 1542 014666 005737 013640
 1543 014672 001064
 1544 014674 042737 000100 177564
 1545 014702 012706 000500
 1546 014706 012737 014776 000064
 1547 014714 005037 000066
 1548 014720 105737 177564
 1549 014724 100375
 1550 014726 012737 000015 177566
 1551 014734 105737 177564
 1552 014740 100375
 1553 014742 012737 000015 177566
 1554 014750 052737 000100 177564
 1555 014756 005037 177776
 1556 014762 000001
 1557 014764 012737 000320 000302
 014772 005212
 014774 000000

 1558 014776 005737 177776
 1559 015002 001405
 015004 012737 000321 000302
 015012 005212
 015014 000000

 1560 015016 023727 000474 014764
 1561 015024 001405
 015026
 015026 012737 000322 000302
 015034 005212
 015036 000000

 1562 015040 005037 177564
 1563 015044
 1564

```

015044 005237 000304
015050 022737 000113 000304
015056 001157
1566 015060 012737 015106 000244
1567 015066 013737 000010 015136
1568 015074 012737 015116 000010
1569 015102 170007
1570 015104 000416
1571 015106
1572 015106 013737 015546 015552
1573 015114 000002
1574 015116
1575 015116 005737 000306
1576 015122 001004
1577 015124 012700 017616
1578 015130 004737 020000
1579 015134 000002
1580 015136 000000
1581 015140 000000
1582 015142
1583 015142 012737 015232 000004
1584
1585 015150 012737 015202 000010
1586 015156 012700 160000
1587 015162 010637 015140
1588 015166 162737 000004 015140
1589 015174 076020
1590 015176 000000
1591 015200 000432
1592 015202
1593
1594 015202 005737 000306
1595 015206 001004
1596 015210 012700 017661
1597 015214 004737 020000
1598 015220 012703 015442
1599 015224 062716 000002
1600 015230 000002
1601 015232 012703 015472
1602 015236 020637 015140
1603 015242 001406
1604 015244 012700 017360
1605 015250 004737 020000
1606 015254 000000
1607 015256 000731
1608 015260 062716 000002
1609 015264 000002
1610 015266 012737 000246 000244
1611 015274 013737 015136 000010
1612 015302 012305
1613 015304 012301
1614 015306 020537 015552
1615 015312 001525
  
```

```

.SBTTL TEST #113 - TEST THAT ALL RESERVED INS TRAP
:*****
:TEST 113 - TEST THAT ALL RESERVED INS TRAP
:*****
TST113: INC $TESTN ;UPDATE TEST NUMBER
        CMP #113,$TESTN ;SEQUENCE ERROR?
        BNE RET4 ;BR TO ERROR HALT ON SEQ ERROR
        MOV #TRAP244,244 ; SET UP TO SEE IF
        MOV 10,TENSAV ; THIS PROCESSOR HAS THE
        MOV #TRAP10,10 ; FLOATING POINT OPTION
        .WORD 170007 ; AN ILLEGAL FPP INSTRUCTION
        BR TSFCIS
TRAP244: ; IF FPP IN--
        MOV FPP,FINISH ; RESET END OF TABLE PCINTER
        RTI ; AND RETURN
TRAP10: ; LEAVE THE TABLE ALONE
        TST $PASS ;FIRST PASS??
        BNE 1$ ;BRANCH IF NO
        MOV #MSGNFP,RO
        JSR PC,PRMSG ;PRINT MESSAGE POINTED TO BY RO
        RTI ; RETURN
1$: TENSAV: .WORD 0 ; A PLACE TO STORE CONTENTS OF 10
        SAVSTK: .WORD 0 ;LOCATION TO SAVE STACK PCINTER
        TSFCIS: ;SEE IF PROCESSOR HAS CIS OPTION
        MOV #AROUND,4 ;SET TIME OUT TRAP VECTOR
        MOV #TNCIS,10 ;SET UP RESERVE INST TRAP VECTOR
        MOV #160000,RO ;POINT RO TO NON-EXISTED MEMORY LOC.
        MOV SP,SAVSTK ;MOVE STACK PCINTER VALUE TO SAVSTK AND
        SUB #4,SAVSTK ;CORRECT TO SHOW 2 EXPECTED PUSHES
        .WORD 76020 ;CIS INST = L2DR (DESTROYS CONTENTS OF RO,R1,R2,R3)
        HALT ;CIS INST FAILED TO TRAP
        BR ADJNC
TNCIS: ;NO CIS OPTION,EXPECTED TRAP ETHER TO 4 OR 10 DID NOT HAPPEN
        TST $PASS ;FIRST PASS
        BNE 1$
        MOV #MSGNCIS,RO ;PRINT MESSAGE POINTED TO BY RO
        JSR PC,PRMSG
1$: MOV #TABLE1,TAB ;CORRECT PCINTER TO AFTER THE HALT
        ADD #2,(SP) ;RETURN
        RTI
AROUND: MOV #TABLE,TAB ;CIS OPTION PRESENT
        CMP SP,SAVSTK ;SEE IF 2 WORDS WERE PUSHED ON THE STACK
        BEQ 1$ ;BRANCH IF SO - CIS SWITCH IN NORMAL POSITION
        MOV #CISMSG,RO ;MOVE ADDR OF CIS SWITCH IN MAINTENANCE POSITION TO RO
        JSR PC,PRMSG ;GO PRINT THIS MESSAGE
        HALT ;HALT, ALLOWING USER TO PUT SWITCH IN NORMAL POSITION
        BR ;GO BACK TO CHECK AGAIN
1$: ADD #2,(SP) ;CORRECT PCINTER TO AFTER THE HALT
        RTI ;RETURN
ADJNC: MOV #246,244 ; RESTORE THE TRAP VECTOR
        MOV TENSAV,10 ; RESTORE THE ILLEGAL INST. VECTOR
GIN1: MOV (TAB)+,FIRST ;FIRST OR CURRENT INSTRUCTION
        MOV (TAB)+,LAST ;LAST INSTRUCTION OR GROUP
        CMP FIRST,FINISH ;TESTED ALL
        BEQ GIN3 ;YES BRANCH
  
```

TEST #113 - TEST THAT ALL RESERVED INS TRAP

```

1616 015314 010537 015554      MOV    FIRST,INST      ;SET UP INST
1617 015320 005237 015554      INC    INST
1618 015324 012737 015346 000010  GIN2:  MOV    #RET,10      ;SET UP RETURN FROM TRAP
1619 015332 012706 000500      MOV    #BUFF,SP       ;SET UP STACK POINTER
1620 015336 005037 177776      CLR    CC              ;CLEAR PRIORITY
1621 015342 000137 015554      JMP    INST            ;EXECUTE RESERVED INSTRUCTION
1622
1623
1624 015346 020627 000474      RET:   CMP    SP,#BUFF-4  ;TEST DECREMENT OF SP
1625 015352 001405      BEQ    RET1
1626 015354 012737 000323 000302  MOV    #323,$FATAL     ;MOVE TO MAILBOX # ***** 323 *****
1626 015362 005212      INC    (R2)            ;SET MSGTYP TO FATAL ERROR
1626 015364 000000      HALT                  ;WRONG DECREMENT
                            ;TO SCOPE REPLACE HALT WITH 240
                            ;AND REPLACE NEXT INST WITH 634
1627 015366 023727 000474 015556  RET1:  CMP    BUFF-4,#INST+2 ;LOC OF INST UNINCREMENTED
1628 015374 001405      BEQ    RET2
1629 015376 012737 000324 000302  MOV    #324,$FATAL     ;MOVE TO MAILBOX # ***** 324 *****
1629 015404 005212      INC    (R2)            ;SET MSGTYP TO FATAL ERROR
1629 015406 000000      HALT                  ;INST INC ON TRAP
                            ;TO SCOPE REPLACE HALT WITH 240
                            ;AND REPLACE NEXT INST WITH 623
1630 015410 005737 000476      RET2:  TST    BUFF-2
1631 015414 001405      BEQ    RET3
1631 015416
1631 015416 012737 000325 000302  RET4:  MOV    #325,$FATAL     ;MOVE TO MAILBOX # ***** 325 *****
1631 015424 005212      INC    (R2)            ;SET MSGTYP TO FATAL ERROR
1631 015426 000000      HALT                  ;CONDITION CODES SET ON TRAP OR WRONG $STNM
                            ;TO SCOPE REPLACE HALT WITH 240
                            ;AND REPLACE NEXT INST WITH 613
1632 015430 023701 015554      RET3:  CMP    INST, LAST
1633 015434 001722      BEQ    GIN1            ;SET UP NEW GROUP
1634 015436 000137 015320      JMP    GIN2            ;FINISH OLD GROUP
1635
1636 015442 076017      TABLE1: 76017          ;CIS INSTRUCTIONS
1637 015444 076032          76032
1638 015446 076037          76037
1639 015450 076045          76045
1640 015452 076047          76047
1641 015454 076077          76077
1642 015456 076117          76117
1643 015460 076132          76132
1644 015462 076137          76137
1645 015464 076145          76145
1646 015466 076147          76147
1647 015470 076177          76177
1648 015472 000007      TABLE: 7
1649 015474 000077          77
1650 015476 000207          207          ;RTS,RT1,JMP
1651 015500 000227          227
1652 015502 007077          7077
1653 015504 007777          7777
1654 015506 075037          075037
1655 015510 076017          76017
1656 015512 076032          76032
1657 015514 076037          76037
1658 015516 076045          76045

```

1659 015520 076047
1660 015522 076132
1661 015524 076137
1662 015526 076145
1663 015530 076147
1664 015532 076077
1665 015534 076117
1666 015536 106377
1667 015540 106477
1668 015542 106677
1669 015544 107777
1670 015546 167777
1671 015550 177777
1672 015552 015552
1673 015554 000000
1674 015556 000000
1675 015560 000000
1676 015562 000000
1677 015564 000000
1678
1679 015566

76047
76132
76137
76145
76147
76077
76117
106377
106477
106677
107777
167777
177777
.
HALT
HALT
HALT
HALT
HALT

FPP:

FINISH:

INST:

GIN3:

; START OF THE FPP INSTRUCTIONS

;END FLAG

;WILL CONTINUE RESERVED INST

;SHOULD TRAP TO LOC 10

;LOC 10 SHOULD SEND YOU TO

;RET

1680

.SBTTL TEST #114 - TEST ILLEGAL HALT
 :*****
 :TEST 114 - TEST ILLEGAL HALT
 :*****

015566 005237 000304
 015572 022737 000114 000304
 015600 001073
 1681 015602 012706 000500
 1682 015606 005037 177766
 1683 015612 012737 015646 000004
 1684 015620 052737 040000 177776
 1685 015626 000000
 1686
 1687 015630 105037 177777
 1688 015634 012737 000326 000302
 015642 005212
 015644 000000

TST114: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #114,\$TESTN ;SEQUENCE ERROR?
 BNE CERIH ;BR TO ERROR HALT ON SEQ ERROR
 MOV #BUFF,SP ;STACK POINTER SETUP
 CLR CPUERR ;CLEAR CPU ERROR REGISTER
 MOV #18,RTRAPS ;SETUP TRAP RETURN
 BIS #040000,PSW ;GO TO SUPER MODE
 HALT ;EXECUTE INST UNDER TEST
 ;FAILURE, NO TRAP
 CLRB PSW+1 ;GO BACK TO KERNEL
 MOV #326,\$FATAL ;MOVE TO MAILBOX # ***** 326 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;HALT IN SUPER MODE FAILED TO TRAP
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 755

1689 015646
 1690 015646 013737 177766 000502
 1691 015654 042737 177413 000502
 1692 015662 022737 000200 000502
 1693 015670 001405
 015672 012737 000327 000302
 015700 005212
 015702 000000

1\$:
 MOV CPUERR,RCPUER ;READ AND SAVE CPU ERROR REGISTER
 BIC #CERMSK,RCPUER ;MASK OFF UNUSED CPU ERR REG BITS
 CMP #200,RCPUER ;IS ILLEGAL HALT BIT SET?
 BEQ 2\$
 MOV #327,\$FATAL ;MOVE TO MAILBOX # ***** 327 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERR REG CONTENTS
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 736

1694 015704 005037 177766
 1695 015710 012737 015744 000004
 1696 015716 052737 140000 177776
 1697 015724 000000
 1698
 1699 015726 105037 177777
 1700 015732 012737 000330 000302
 015740 005212
 015742 000000

2\$:
 CLR CPUERR ;CLEAR CPU ERR REG
 MOV #38,RTRAPS ;SETUP TRAP RETURN
 BIS #140000,PSW ;GO TO USER MODE
 HALT
 ;FAILURE, NO TRAP
 CLRB PSW+1 ;GO BACK TO KERNEL
 MOV #330,\$FATAL ;MOVE TO MAILBOX # ***** 330 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;HALT IN USER MODE FAILED TO TRAP
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 716

1701 015744
 1702 015744 013737 177766 000502
 1703 015752 042737 177413 000502
 1704 015760 022737 000200 000502
 1705 015766 001405
 015770
 015770 012737 000331 000302
 015776 005212
 016000 000000

3\$:
 MOV CPUERR,RCPUER ;SAVE CPU ERROR REGISTER
 BIC #CERMSK,RCPUER ;MASK OFF UNUSED CPU ERR REG BITS
 CMP #200,RCPUER ;IS ILLEGAL HALT BIT SET?
 BEQ DONE
 CERIH:
 MOV #331,\$FATAL ;MOVE TO MAILBOX # ***** 331 *****
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;INCORRECT CPU ERR REG CONTENTS
 ;TO SCOPE REPLACE HALT WITH 240
 ;AND REPLACE NEXT INST WITH 677

1706 016002
 1707 016002 005037 177766
 1708 016006 105037 177777

DONE:
 CLR CPUERR
 CLRB PSW+1 ;GO BACK TO KERNEL MODE

1709

.SBTTL TEST #115 - TEST SPL INST. FOR NOP IN USER/SUPER MODES
 :*****
 :TEST 115 - TEST SPL INST. FOR NOP IN USER/SUPER MODES
 :*****

016012	005237	000304			TST115: INC	\$TESTN	;UPDATE TEST NUMBER
016016	022737	000115	000304		CMP	#115,\$TESTN	;SEQUENCE ERROR?
016024	001125				BNE	SEQ	;BR TO ERROR HALT ON SEQ ERROR
1710 016026	012706	000500			MOV	#BUFF,SP	;SETUP STACK
1711 016032	052737	040000	177776		BIS	#040000,PSW	;GO TO SUPER MODE
1712 016040	000277				SCC		;SET CC
1713 016042	000231				SPL	1	;SPL SHOULD=NOP IN USER/SUPER MODES
1714 016044	000232				SPL	2	
1715 016046	000233				SPL	3	
1716 016050	000234				SPL	4	
1717 016052	000235				SPL	5	
1718 016054	000236				SPL	6	
1719 016056	000237				SPL	7	
1720 016060	013737	177776	016742		MOV	PSW,SPSW	;SAVE PSW
1721 016066	023727	016742	040017		CMP	SPSW,#040017	;VERIFY THAT PSW HAS NOT CHANGED
1722 016074	001407				BEQ	1\$	
1723 016076	105037	177777			CLRB	PSW+1	;GO BACK TO KERNEL
1724 016102	012737	000332	000302		MOV	#332,\$FATAL	;MOVE TO MAILBOX # ***** 332 *****
016110	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
016112	000000				HALT		;PRIORITY LEVELS CHANGE
							;TO SCOPE REPLACE HALT WITH 240
							;AND REPLACE NEXT INST WITH 744
1725 016114	012737	040340	177776	1\$:	MOV	#040340,PSW	;SET PRIORITY TO 7
1726 016122	000257				CCC		;CLEAR CONDITION CODES
1727 016124	000230				SPL	0	;SPL SHOULD=NOP IN SUPERVISOR MODE
1728 016126	013737	177776	016742		MOV	PSW,SPSW	;SAVE PSW
1729 016134	023727	016742	040340		CMP	SPSW,#040340	;VERIFY THAT PSW PRIORITY AND CONDITION CODES HAVE NOT CHANGE
1730 016142	001407				BEQ	2\$	
1731 016144	105037	177777			CLRB	PSW+1	;GO BACK TO KERNEL
1732 016150	012737	000333	000302		MOV	#333,\$FATAL	;MOVE TO MAILBOX # ***** 333 *****
016156	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
016160	000000				HALT		;SPL INSTRUCTION CHANGED PSW SHOULD BE NOP
							;TO SCOPE REPLACE HALT WITH 240
							;AND REPLACE NEXT INST WITH 721
1733 016162	012737	140000	177776	2\$:	MOV	#140000,PSW	;GO TO USER MODE
1734 016170	000277				SCC		;SET CC
1735 016172	000231				SPL	1	;SPL SHOULD=NOP IN USER MODE
1736 016174	000232				SPL	2	
1737 016176	000233				SPL	3	
1738 016200	000234				SPL	4	
1739 016202	000235				SPL	5	
1740 016204	000236				SPL	6	
1741 016206	000237				SPL	7	
1742 016210	013737	177776	016742		MOV	PSW,SPSW	;SAVE PSW
1743 016216	023727	016742	140017		CMP	SPSW,#140017	;VERIFY THAT PSW HAS NOT CHANGED
1744 016224	001407				BEQ	3\$	
1745 016226	105037	177777			CLRB	PSW+1	;GO BACK TO KERNEL
1746 016232	012737	000334	000302		MOV	#334,\$FATAL	;MOVE TO MAILBOX # ***** 334 *****
016240	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
016242	000000				HALT		;PRIORITY LEVELS HAS CHANGED
							;TO SCOPE REPLACE HALT WITH 240

```

1747 016244 012737 140340 177776 3$: MOV #140340,PSW ;AND REPLACE NEXT INST WITH 670
1748 016252 000257 CCC ;SET PRIORITY TO 7
1749 016254 000230 SPL 0 ;CLEAR CONDITION CODES
1750 016256 013737 177776 016742 MOV PSW,SPSW ;SPL SHOULD=NOP IN USER MODE
1751 016264 023727 016742 140340 CMP SPSW,#140340 ;SAVE PSW
1752 016272 001407 BEQ FSPL ;VARIIFY THAT PSW PRIORITY AND CONDITION CODES HAVE NOT CHANGE
1753 016274 105037 177777 CLRB PSW+1 ;GO BACK TO KERNEL
1754 016300 SEQ:
    016300 012737 000335 000302 MOV #335,$FATAL ;MOVE TO MAILBOX # ***** 335 *****
    016306 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016310 000000 HALT ;SPL INST.CHANGED PSW OR WRONG TEST#
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 645

1755 016312 FSPL:
1756 016312 105037 177777 CLRB PSW+1 ;GO BACK TO KERNEL
1757 ;THIS TEST VARIFIES FOR ALL COMBINATIONS OF PIR AND PROCESSOR
1758 ;PRIORITY LEVELS THAT REQUESTS ARE GRANTED (TRAP TO 240 OCCURS
1759 ;BY THE PROCESSOR,ONLY WHEN THE PIR IS AT A HIGHER LEVEL THAN
1760 ;THE PROCESSOR.
1761 ;THE CONTENTS OF SPIR,SPSW AND TRP240 SHOULD BE EXAMINED ON ERROR.
1762 ;SPIR BITS 2-0 CONTAINS ONE LESS THAN THE PIR REQUEST LEVEL AT
1763 ;THE TIME OF ERROR.SPSW BITS 2-0 CONTAINS THE PROCESSOR PRIORITY
1764 ;AT THE TIME OF ERROR.TRP240 INDICATES WHETHER OR NOT A TRAP WAS
1765 ;EXPECTED (1= EXPECTING TRAP TO 240)
1766 ;THE SPL INSTRUCTION IS USED TO SETUP PROCESSOR PRIORITY.
1767 ;NOTE: THIS IS THE FIRST REAL TEST OF THE SPL INST.
1768 ;ON ERROR,IF EXPECTED PIRQ TRAP DID NOT OCCURE VARIFY SPL
1769 ;OPERATION BY COMPARING SPSW BITS<2-0> WITH PSW PRIORITY
1770 ;BITS<7-5>.ON ERROR IF AN UNSPECTED PIRQ TRAP OCCURED
1771 ;VARIFY SPL OPERATION BY COMPARING SPSW BITS<2-0> WITH
1772 ;PROCESSOR PSW<7-5> ON STACK.
1773

```

1774

```

.SBTTL TEST #116 - TEST PIRQ LEVELS AND SPL INSTRUCTION
:*****
:TEST 116 - TEST PIRQ LEVELS AND SPL INSTRUCTION
:*****
TST116: INC $TESTN ;UPDATE TEST NUMBER
        CMP #16,$TESTN ;SEQUENCE ERROR?
        BNE PTRP ;BR TO ERROR HALT ON SEQ ERROR
        CLR PIRPSW
        MOV #2TRP,0 ;SET LOCATION ZERO TRAP VEC
        MOV #340,2
        MOV #T4TRP,4 ;SET UP FAILURE TRAP VEC
        MOV #340,6
        MOV #T24TRP,24 ;SETUP POWER FAIL VEC
        MOV #340,26
        MOV #PQTRP,PIRVC1 ;SETUP PIRQ VEC
        MOV #340,PIRVC2 ;SET 242 TO PRIORITY 7
MLOOP: MOV #BUFF,SP ;SETUP STACK
        BIS #340,PSW ;SET PROCESSOR PRIORITY TO 7
        ;COMPUTE EXPECTED RESULT
        MOV PIRPSW,R0
        BIC #177707,R0 ;SETUP SPIR
        ASR R0
        ASR R0
        ASR R0
        MOV R0,SPIR ;SAVE R0 IN SPIR
        MOV PIRPSW,R0
        BIC #177770,R0 ;SETUP SPSW
        MOV R0,SPSW
        CMP SPIR,SPSW
        BGE 2$ ;BRANCH IF PIR > PSW
        CLR TRP240 ;CLEAR FLAG
        BR 3$
        MOV #177777,TRP240 ;SET FLAG
        JSR PC,SETPIRQ ;SETUP PIRQ BASE ON THE # IN SPIR
        ;SET PSW PRIORITY BASE ON THE # IN SPSW
        BIC #7,1$ ;CLEAR LSB 3 BITS OF SPL INST.AT 1$
        BIS SPSW,1$ ;SET LSB 3 BITS OF SPL INST.TO DESIRED PROC. PRI.
        SPL 0 ;THE ACTUAL PRIORITY SET INTO THE PSW IS CONTROLLED BY
        ;THE PREVIOUS TWO INSTRUCTION.
        NOP
        TST TRP240
        BEQ NXTST
        MOV #336,$FATAL ;MOVE TO MAILBOX # ***** 336 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;EXPECTED PIRQ TRAP BUT DID NOT GET IT
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 717
        ;SET UP FOR PASS COUNT
NXTST: NOP
        INC PIRPSW ;SETUP FOR NEXT PASS THROUGH LOOP
        CMP PIRPSW,#70
        BLT MLOOP ;BRANCH TO MLOOP IF COUNT IS LESS THAN 70
        CLR PIRQ ;DONE WITH LOOPING PIRQ REQUESTS
        MOV #6,4 ;RESTORE RETURNS
    
```

016316 005237 000304
 016322 022737 000116 000304
 016330 001154
 1775 016332 005037 016744
 1776 016336 012737 016676 000000
 1777 016344 012737 000340 000002
 1778 016352 012737 016712 000004
 1779 016360 012737 000340 000006
 1780 016366 012737 016726 000024
 1781 016374 012737 000340 000026
 1782 016402 012737 016640 000240
 1783 016410 012737 000340 000242
 1789 016416 012706 000500
 1790 016422 052737 000340 177776
 1791
 1792 016430 013700 016744
 1793 016434 042700 177707
 1794 016440 006200
 1795 016442 006200
 1796 016444 006200
 1797 016446 010037 016740
 1798 016452 013700 016744
 1799 016456 042700 177770
 1800 016462 010037 016742
 1801 016466 023737 016740 016742
 1802 016474 002003
 1803 016476 005037 016746
 1804 016502 000403
 1805 016504 012737 177777 016746 2\$:
 1806 016512 004737 016750 3\$:
 1807
 1808 016516 042737 000007 016532
 1809 016524 053737 016742 016532
 1810 016532 000230 1\$:
 1811
 1812 016534 000240
 1813 016536 005737 016746
 1814 016542 001405
 016544 012737 000336 000302
 016552 005212
 016554 000000
 1815 016556 000240
 1816
 1817
 1818 016560 005237 016744
 1819 016564 023727 016744 000070
 1820 016572 002711
 1821 016574 005037 177772
 1822 016600 012737 000006 000004

1823	016606	005037	000006		CLR	6		
1824	016612	005037	000000		CLR	0		
1825	016616	012737	017712	000024	MOV	#PWRDWN,24		
1826	016624	012737	000242	000240	MOV	#242,240		
1827	016632	005037	000242		CLR	242		
1828	016636	000542			BR	TST117		
1829	016640	005737	016746		PQTRP: TST	TRP240		;PRIORITY TRAP SERVICE
1830	016644	001344			BNE	NXTST		
	016646	012737	000337	000302	MOV	#337,\$FATAL		;MOVE TO MAILBOX # ***** 337 *****
	016654	005212			INC	(R2)		;SET MSGTYP TO FATAL ERROR
	016656	000000			HALT			;EXPECTED NO PIRQ TRAP,BUT GOT ONE
								;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 656
1831	016660	000736			BR	NXTST		
1835	016662				PTRP: MOV	#340,\$FATAL		;MOVE TO MAILBOX # ***** 340 *****
	016662	012737	000340	000302	INC	(R2)		;SET MSGTYP TO FATAL ERROR
	016670	005212			HALT			;WRONG TEST NUMBER
	016672	000000						;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 616
1839	016674	000523			BR	TST117		
1840	016676				ZTRP: MOV	#341,\$FATAL		;MOVE TO MAILBOX # ***** 341 *****
1841	016676	012737	000341	000302	INC	(R2)		;SET MSGTYP TO FATAL ERROR
	016704	005212			HALT			;UNSPECED TRAP TO ZERO OCCURED DURING PIRQ TST
	016706	000000						;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 642
1842	016710	000722			BR	NXTST		
1843	016712				T4TRP: MOV	#342,\$FATAL		;MOVE TO MAILBOX # ***** 342 *****
1844	016712	012737	000342	000302	INC	(R2)		;SET MSGTYP TO FATAL ERROR
	016720	005212			HALT			;UNSPECED TRAP TO 4 DURING PIRQ TESTING
	016722	000000						;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 634
1845	016724	000714			BR	NXTST		
1846	016726				T24TRP: MOV	#343,\$FATAL		;MOVE TO MAILBOX # ***** 343 *****
1847	016726	012737	000343	000302	INC	(R2)		;SET MSGTYP TO FATAL ERROR
	016734	005212			HALT			;UNSPECED POWER FAIL TRAP DURING PIRQ TESTING
	016736	000000						;TO SCOPE REPLACE HALT WITH 240
								;AND REPLACE NEXT INST WITH 626
1848	016740	000000			SPIR:	.WORD	0	
1849	016742	000000			SPSW:	.WORD	0	
1850	016744	000000			PIRPSW:	.WORD	0	

1852 177772 PIRQ=177772
 1853 000240 PIRVC1=240
 1854 000242 PIRVC2=242
 1855 016746 000000 TRP240: .WORD 0
 1856
 1857

1858 016750 SETPIRQ:
 1859 016750 023727 016740 000000 CMP SPIR,#0
 1860 016756 001435 BEQ 1\$
 1861 016760 023727 016740 000001 CMP SPIR,#1
 1862 016766 001435 BEQ 2\$
 1863 016770 023727 016740 000002 CMP SPIR,#2
 1864 016776 001435 BEQ 3\$
 1865 017000 023727 016740 000003 CMP SPIR,#3
 1866 017006 001435 BEQ 4\$
 1867 017010 023727 016740 000004 CMP SPIR,#4
 1868 017016 001435 BEQ 5\$
 1869 017020 023727 016740 000005 CMP SPIR,#5
 1870 017026 001435 BEQ 6\$
 1871 017030 023727 016740 000006 CMP SPIR,#6
 1872 017036 001435 BEQ 7\$
 1873 017040 012737 000344 000302 MCV #344,\$FATAL
 017046 005212 INC (R2)
 017050 000000 HALT

:MOVE TO MAILBOX # ***** 344 *****
 :SET MSGTYP TO FATAL ERROR
 :# IN SPIR DOES NOT MAKE SENSE OR SPIR NOT=0-6
 :TO SCOPE REPLACE HALT WITH 240
 :AND REPLACE NEXT INST WITH 561

1874 017052 012737 001000 177772 1\$: MOV #1000,PIRQ
 1875 017060 000430 BR 10\$
 1876 017062 012737 002000 177772 2\$: MOV #2000,PIRQ
 1877 017070 000424 BR 10\$
 1878 017072 012737 004000 177772 3\$: MOV #4000,PIRQ
 1879 017100 000420 BR 10\$
 1880 017102 012737 010000 177772 4\$: MOV #10000,PIRQ
 1881 017110 000414 BR 10\$
 1882 017112 012737 020000 177772 5\$: MOV #20000,PIRQ
 1883 017120 000410 BR 10\$
 1884 017122 012737 040000 177772 6\$: MOV #40000,PIRQ
 1885 017130 000404 BR 10\$
 1886 017132 012737 100000 177772 7\$: MOV #100000,PIRQ
 1887 017140 000400 BR 10\$
 1888 017142 000207 10\$: RTS PC
 1889
 1890

1891 :TABLE FOR PIRQ SETUP
 1892 :SPIR PIR LEVEL SETPIRQ # LOADED INTO PIRQ REG R0
 1893 :0000 1 BIT9 1000
 1894 :0001 2 BIT10 2000
 1895 :0002 3 BIT11 4000
 1896 :0003 4 BIT12 10000
 1897 :0004 5 BIT13 20000
 1898 :0005 6 BIT14 40000
 1899 :0006 7 BIT15 100000

1900

```

.SBTTL TEST #117 - CHK ODD ADRS TRAP WITH SP AT ODD ADRS ON RTI
:*****
:TEST 117 - CHK ODD ADRS TRAP WITH SP AT ODD ADRS ON RTI
:*****
TST117: INC $TESTM ;UPDATE TEST NUMBER
        CMP #117,$TESTM ;SEQUENCE ERROR?
        BNE OATSE ;BR TO ERROR HALT ON SEQ ERROR
        MOV #140000,177776 ;SET CURRENT MODE TO USER
        MOV #1,SP ;PUT ODD ADDRESS IN STACK POINTER
        MOV #1$,4 ;TRAPS TO 1$
        MOV #340,6 ;PRIORITY TO 7
        RTI ;TEST RTI - SHOULD ODD ADDRESS TRAP TO 4
1$:     CMP #30340,177776 ;SEE IF PREVIOUS MODE USER AND PRIORITY 7 LOADED
        BEQ END ;BRANCH OUT IF OK
:*****
:IF SWR CONTAINS 30000, YOU ARE MISSING ECO #6.
:*****
        MOV #345,$FATAL ;MOVE TO MAILBOX # ***** 345 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;PSW NOT PROPERLY LOADED
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 752
        ;BRANCH TO END ROUTINE
OATSE: BR END
        MOV #346,$FATAL ;MOVE TO MAILBOX # ***** 346 *****
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;WRONG $TSTNM
        ;TO SCOPE REPLACE HALT WITH 240
        ;AND REPLACE NEXT INST WITH 744
  
```

```

017144 005237 000304
017150 022737 000117 000304
017156 001026
1901 017160 012737 140000 177776
1902 017166 012706 000001
1903 017172 012737 017210 000004
1904 017200 012737 000340 000006
1905 017206 000002
1906 017210 022737 030340 177776
1907 017216 001413
1908
1909
1910
1911 017220 012737 000345 000302
    017226 005212
    017230 000000
1912 017232 000405
1913 017234
    017234 012737 000346 000302
    017242 005212
    017244 000000
  
```

1915						.SBTTL	END OF PASS ROUTINE	
1916	017246	005237	000306			INC	\$PASS	
1917	017252	105237	017356		END:	INCB	PASSPT	: SHOULD PRINT THIS PASS?
1918	017256	001020				BNE	ACT	: NO
1919	017260	132737	000040	000321		BITB	#40,\$ENVN	: WILL APT ALLOW PRINTING?
1920	017266	001014				BNE	ACT	: NO
1921	017270	023727	000042	017330		CMP	42,#SENDAD	
1922	017276	001410				BEQ	ACT	
1923	017300	012700	017503			MOV	#MSG,RO	: GET MSG ADDR.
1924	017304	004737	020000			JSR	PC,PRMSG	: PRINT MESSAGE POINTED TO BY RO
1925	017310	000005				RESET		
1926	017312	012737	177761	017356		MOV	#177761,PASSPT	: DO IT ABOUT 15 DECIMAL TIMES
1927	017320	013700	000042		ACT:	MOV	42,RO	: CHECK ACT
1928	017324	001405				BEQ	GOAGIN	: KEEP GOING
1929	017326	000005				RESET		
1930	017330	004710			SENDAD:	JSR	PC,(RO)	: ACT HOOKS
1931	017332	000240				NOP		
1932	017334	000240				NOP		
1933	017336	000240				NOP		
1934	017340	012737	000012	000010	GOAGIN:	MOV	#12,10	
1935	017346	005037	000012			CLR	12	
1936	017352	000137	000654			JMP	RESTR	: DO NEXT PASS
1937	017356	177777			PASSPT:	-1		

CKKABBO 11/44 TRAPS
ASCII MESSAGES

MACRO M1113 09-OCT-80 16:58 PAGE 94 SEQUENCE 112

1 9

1938				.SBTTL	ASCII MESSAGES
1939 017360	015	012	103	CISMSG: .ASCII	<15><12>.CIS SWITCH IS IN THE MAINTENANCE POSITION.<15><12>
1940 017435	120	125	124	.ASCIZ	.PUT IN NORMAL POSITION AND CONTINUE.<15><12>
1941 017503	015	012	105	MSG: .ASCIZ	<15><12>.END OF CKKABBO 11/44 TRAPS.
1942 017540	015	012	103	PNAME: .ASCIZ	<15><12>+CKKABBO 11/44 TRAPS+
1943 017566	015	012	103	TITLE: .ASCIZ	<15><12>+CKKABBO 11/44 TRAPS+<15><12>
1944 017616	015	012	116	MSGNFP: .ASCIZ	<15><12>.NO FLOATING POINT OPTION PRESENT.
1945 017661	015	012	116	MSGNCIS: .ASCIZ	<15><12>.NO CIS OPTION PRESENT .
1946				.EVEN	


```

1947
1948 017712 012737 017722 000024 PWRDWN: .SBTTL POWER DOWN AND POWER UP ROUTINES
1949 017720 000000                MOV      #PWRUP,24
1950                HALT
1951 017722 012737 017712 000024 PWRUP:  MOV      #PWRDWN,24
1952 017730 012706 000500        MOV      #BUFF,SP
1953 017734 132737 000040 000321    BITB    #40,$ENVM      ;WILL APT ALLOW PRINTING?
1954 017742 001004                BNE     PFRES          ;NO
1955 017744 012700 017760        MOV      #MSGPWF,RO    ;GET MSG ADDR.
1956 017750 004737 020000        JSR     PC,PRTMSG     ;PRINT MESSAGE POINTED TO BY RO
1957 017754 000137 000654        PFRES:  JMP     RESTR
1958 017760    015    012    120  MSGPWF: .ASCIZ <15><12>.POWER FAILED!.
    
```

```
1959  
1960 020000 132737 000040 000321 PRTMSG: .SBTTL SUBROUTINE TO PRINT A MESSAGE  
1961 020006 001011                    BITB #40,$ENVM        ;WILL APT ALLOW PRINTING?  
1962 020010 105737 177564            BNE 1$            ;BRANCH IF NO  
1963 020014 100375                    TSTB TPS           ;TTY READY  
1964 020016 112037 177566            BPL 2$            ;NO WAIT  
1965 020022 001372                    MOVB (R0)+,TPB     ;PRINT CHARACTER  
1966 020024 105737 177564            BNE 2$            ;NEXT IF NOT DONE.  
1967 020030 100375                    TSTB TPS  
1968 020032 000207                    BPL 3$  
1969           000001                    RTS PC  
                                      .END
```

CKKABBO 11/44 TRAPS
SYMBOL TABLE

ABASE = 000000
 ACDW1 = 000000
 ACDW2 = 000000
 ACPUOP= 000000
 ACT = 017320
 ADALL = 013236
 ADDW0 = 000000
 ADDW1 = 000000
 ADDW10= 000000
 ADDW11= 000000
 ADDW12= 000000
 ADDW13= 000000
 ADDW14= 000000
 ADDW15= 000000
 ADDW2 = 000000
 ADDW3 = 000000
 ADDW4 = 000000
 ADDW5 = 000000
 ADDW6 = 000000
 ADDW7 = 000000
 ADDW8 = 000000
 ADDW9 = 000000
 ADEVCT= 000000
 ADEVM = 000000
 ADJNC = 015266
 ADREND 000620
 ADRTAB 000520
 AENV = 000000
 AENVM = 000000
 AFATAL= 000000
 AMADR1= 000000
 AMADR2= 000000
 AMADR3= 000000
 AMADR4= 000000
 AMAMS1= 000000
 AMAMS2= 000000
 AMAMS3= 000000
 AMAMS4= 000000
 AMSGAD= 000000
 AMSGLG= 000000
 AMSGTY= 000000
 AMTYP1= 000000
 AMTYP2= 000000
 AMTYP3= 000000
 AMTYP4= 000000
 APASS = 000000
 APRIOR= 000000
 AROUND 015232
 ASWREG= 000000
 ATESTN= 000000
 ATRAP = 013304
 AUNIT = 000000
 AUSWR = 000000
 AUTO = 013272
 AUTO1 = 013570
 AVECT1= 000000
 AVECT2= 000000

BEGIN = 000622
 BELL = 000240
 BR1 = 001030
 BR10 = 001262
 BR11 = 001376
 BR12 = 001446
 BR13 = 001516
 BR14 = 001566
 BR15 = 001674
 BR16 = 001716
 BR17 = 001740
 BR2 = 001060
 BR20 = 001762
 BR21 = 002004
 BR22 = 002026
 BR23 = 002050
 BR3 = 001104
 BR33 = 002130
 BR34 = 002144
 BR35 = 002160
 BR36 = 002174
 BR37 = 002220
 BR4 = 001132
 BR40 = 002234
 BR41 = 002250
 BR45 = 003750
 BR46 = 004600
 BR46A = 004612
 BR47 = 005460
 BR5 = 001160
 BR51 = 006310
 BR51A = 006322
 BR6 = 001206
 BR60 = 012662
 BR7 = 001234
 BR70 = 013072
 BR71 = 014126
 BTRAP = 013450
 BUFF = 000500
 CC = 177776
 CERIH = 015770
 CERMSK= 177413
 CERR1 = 010054
 CERR2 = 010734
 CISMSG = 017360
 CORH = 013234
 CPUERR= 177766
 CTRAP = 013404
 DL11W = 013642
 DL11W1 = 013646
 DONE = 016002
 DTRAP = 013424
 END = 017246
 ERRP1 = 010042
 ERRP2 = 010722
 FINISH = 015552
 FIRST = X000005

FOVER = 011750
 FPP = 015546
 FSPL = 016312
 GIN1 = 015302
 GIN2 = 015320
 GIN3 = 015566
 GOAGIN = 017340
 HERE = 000000
 HLT = 000000
 ILLA = 004700
 ILLB = 000100
 INST = 015554
 INSTC = 002424
 INSTK = 007272
 KPAR0 = 000600
 KPAR1 = 000602
 KPAR2 = 000604
 KPAR3 = 000606
 KPAR4 = 000610
 KPAR5 = 000612
 KPAR6 = 000614
 KPAR7 = 000616
 KPDR0 = 000560
 KPDR1 = 000562
 KPDR2 = 000564
 KPDR3 = 000566
 KPDR4 = 000570
 KPDR5 = 000572
 KPDR6 = 000574
 KPDR7 = 000576
 KTSTA = 000516
 KTVEC = 000514
 K1 = 000742
 K10 = 000760
 K11 = 000762
 K12 = 000764
 K2 = 000744
 K3 = 000746
 K4 = 000750
 K5 = 000752
 K6 = 000754
 K7 = 000756
 LAST = X000001
 MLOOP = 016416
 MSG = 017503
 MSGNCI = 017661
 MSGNFP = 017616
 MSGPWF = 017760
 NODL = 014140
 NODL1 = 014276
 NODL2 = 014402
 NODL3 = 014652
 NODL4 = 015044
 NOP = 000240
 NOR = 013256
 NXTST = 016556
 OATSE = 017234

PASSPT = 017356
 PFRES = 017754
 PIRPSW = 016744
 PIRQ = 177772
 PIRVC1= 000240
 PIRVC2= 000242
 PNAME = 017540
 PQTRP = 016640
 PROFTE = 013640
 PRTMSG = 020000
 PSW = 177776
 PTRP = 016662
 PWRDWN = 017712
 PWRUP = 017722
 RA = 005472
 RA1 = 003762
 RB = 005456
 RB1 = 003746
 RC = 005452
 RCPUER = 000502
 RC1 = 003742
 RESET2 = 014460
 RESET3 = 014446
 RESTRT = 000654
 RET = 015346
 RETA = 002314
 RETAH = 002326
 RETAT = 012050
 RETA1 = 003154
 RETA2 = 004052
 RETA3 = 004666
 RETA4 = 005562
 RETA5 = 006376
 RETB = 002356
 RETBT = 012124
 RETB1 = 003204
 RETB2 = 004102
 RETB3 = 004716
 RETB4 = 005612
 RETB5 = 006426
 RETC = 002426
 RETCT = 012204
 RETC1 = 003254
 RETC2 = 004152
 RETC3 = 004766
 RETC4 = 005662
 RETC5 = 006476
 RETD = 002506
 RETD1 = 003334
 RETD2 = 004232
 RETD3 = 005046
 RETD4 = 005742
 RETD5 = 006556
 RETE = 002554
 RETE1 = 003400
 RETE2 = 004300
 RETE3 = 005114

RETE4 = 006010
 RETE5 = 006624
 RETF = 002632
 RETF1 = 003456
 RETF2 = 004356
 RETF3 = 005172
 RETF4 = 006066
 RETF5 = 006702
 RETG = 002756
 RETG1 = 003602
 RETG2 = 004502
 RETG3 = 005316
 RETG4 = 006212
 RETG5 = 007026
 RETH5 = 007174
 RETJ = 007224
 RETK = 007214
 RETL = 007354
 RETM = 007422
 RETN = 007500
 RETO = 007622
 RETP = 010016
 RETQ = 010112
 RETR = 010164
 RETS = 010246
 RETT = 010316
 RETU = 010376
 RETV = 010524
 RET1 = 015366
 RET2 = 015410
 RET3 = 015430
 RET4 = 015416
 RTI1 = 012460
 RTI2 = 012474
 RTRAP = 000010
 RTRAP1= 000034
 RTRAP2= 000020
 RTRAP3= 000030
 RTRAP4= 000014
 RTRAP5= 000004
 RTT1 = 012270
 RTT2 = 012306
 RTT3 = 012354
 RTT4 = 012374
 RTT5 = 012326
 RTT6 = 012416
 R6 = X000006
 R7TRX = 014002
 R7TR1 = 012556
 R7TR2 = 012626
 R7TR2A = 012614
 R7TR3 = 012676
 R7TR4 = 012750
 SAVSTK = 015140
 SEQ = 016300
 SETPIR = 016750
 SPIR = 016740

CKKAB90 11/44 TRAPS
SYMBOL TABLE

SPSW	016742	TR4A	012736	TST32	005422	TST77	012416	WATE2	014734
SRO	000504	TR5	014236	TST33	005520	TTCSR =	177564	WATE3	014762
SROM	000506	TSFCIS	015142	TST34	005562	TTT37	014646	WATE4	015040
SR1	000510	TSL	013232	TST35	005632	TTY11	014634	WATE5	015026
SR2	000512	TST1	000766	TST36	005704	TTY3	014552	ZTRP	016676
STATUS=	177776	TST10	002450	TST37	006032	TTY4	014626	\$APTHD	000330
STP	011762	TST100	012512	TST4	002072	T24TRP	016726	\$CPUOP	000326
STPP	003054	TST101	013000	TST40	006334	T4TRP	016712	\$DEVCT	000310
STPPA	003066	TST102	013120	TST41	006376	UPAR0	000540	\$ENDAD	017330
STP3	013174	TST103	013214	TST42	006446	UPAR1	000542	\$ENV	000320
STP3D	013206	TST104	013654	TST43	006520	UPAR2	000544	\$ENVM	000321
TAB	=X000003	TST105	014002	TST44	006646	UPAR3	000546	\$ERN =	000347
TABLE	015472	TST106	014140	TST45	007132	UPAR4	000550	\$ERROR=	000302
TABLE1	015442	TST107	014276	TST46	007174	UPAR5	000552	\$ETABL	000320
TDEC1	010676	TST11	002576	TST47	007244	UPAR6	000554	\$ETEND	000330
TDEC2	010770	TST110	014402	TST5	002264	UPAR7	000556	\$FATAL	000302
TDEC3	011044	TST111	014476	TST50	007316	UPDR0	000520	\$HIBTS	000330
TDEC4	011116	TST112	014652	TST51	007444	UPDR1	000522	\$MAIL	000300
TDEC6	011132	TST113	015044	TST52	007726	UPDR2	000524	\$MBADR	000332
TDEC7	011144	TST114	015566	TST53	010060	UPDR3	000526	\$MSGAD	000314
TDEC77	013764	TST115	016012	TST54	007132	UPDR4	000530	\$MSGLG	000316
TDEC8	013752	TST116	016316	TST55	007206	UPDR5	000532	\$MSGTY	000300
TENSAV	015136	TST117	017144	TST56	010340	UPDR6	000534	\$PASS	000306
TITLE	017566	TST12	003100	TST57	010630	UPDR7	000536	\$PASTM	000336
TNCIS	015202	TST13	003154	TST6	002326	VDEC1	011214	\$SVPC =	000300
TONT1	013104	TST14	003224	TST60	010740	VDEC10	011472	\$SWR =	000000
TPB	= 177566	TST15	003276	TST61	011010	VDEC11	011550	\$SWREG	000322
TPS	= 177564	TST16	003422	TST62	011144	VDEC12	011562	\$TESTN	000304
TRACE	013060	TST17	003712	TST63	011222	VDEC13	011646	\$TN =	000120
TRAPA =	000010	TST2	001312	TST64	011300	VDEC14	011660	\$TSTM	000334
TRAPB	013602	TST20	004010	TST65	011356	VDEC2	011202	\$TSTM=	000304
TRAP10	015116	TST21	004052	TST66	011434	VDEC3	011272	\$UNIT	000312
TRAP24	015106	TST22	004122	TST67	011512	VDEC4	011260	\$UNITM	000340
TRCSR =	177560	TST23	004174	TST7	002376	VDEC5	011350	\$USWR	000324
TRC1	013164	TST24	004322	TST70	011610	VDEC6	011336	\$X =	017160
TRP240	016746	TST25	004624	TST71	011666	VDEC7	011426	\$XX =	177745
TRT =	000003	TST26	004666	TST72	011774	VDEC8	011414	\$XXX =	000744
TRO	014102	TST27	004736	TST73	012050	VDEC9	011504	\$Y =	016332
TR2	014114	TST3	001636	TST74	012144	WATE	014776	\$YY =	016416
TR3	014234	TST30	005010	TST75	012226	WATE1	014720	.\$X =	000330
TR4	014250	TST31	005136	TST76	012306				

. ABS. 020034 000
000000 001
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 7800 WORDS (31 PAGES)
DYNAMIC MEMORY: 8730 WORDS (33 PAGES)
ELAPSED TIME: 00:03:45
CKKABB.BIN,CKKABB.SEQ/-SP/NL:TOC=CKKABB.MLB/ML,CKKABB.P11