

KD11-Z

11/44 TRAPS
CKKABCO

AH-F623C-MC
FICHE 1 OF 1

AUG 1981
COPYRIGHT © 79-81
MADE IN USA



A microfiche card containing a grid of 11 columns and 44 rows of frames. Each frame contains a small, high-contrast image, likely a trap or a specific data point. The frames are arranged in a regular grid pattern across the card.

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

.REM %

IDENTIFICATION

PRODUCT CODE:	AC-F621C-MC
PRODUCT NAME:	CKKABCO 11/44 TRAPS
DATE CREATED:	APRIL, 1981
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	DAN MILLEVILLE

COPYRIGHT (C) 1979, 1981 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.

34
35
36
37
38

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.

39
40
41
42
43
44
45
46
47
48

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.

49
50
51
52
53

3. LOADING PROCEDURE

3.1 METHOD

PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.

54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69

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

4.1 PROGRAM AND/OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY. (BOTTOM 4K).

LOAD ADDRESS.

START.

THE PROGRAM WILL LOOP.

IT WILL PRINT "CKKABCO 11/44 TRAPS" AFTER THE FIRST ITERATION
AND THEN PRINTS IT EVERY 15 TIMES (APPROXIMATELY EVERY 15 SECONDS)

70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114

5. OPERATION

5.1 SUBROUTINE ABSTRACTS

5.1.1 BEGIN AT 200

5.1.2 SCOPE

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

5.1.3 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 CON-
TAIN A HALT (00000) (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.

5.2 PROGRAM AND/OR OPERATOR ACTION

5.2.1 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.

115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158

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 BRANCH TO A HALT
**OR:
        BEQ   1000$       ;BRANCH AROUND JUMP IF OK
        JMP   TSTA+1-12   ;JUMP (USED FOR LONG TESTS THE BRANCH CAN'T REACH)
1000$:
        ----
        CODE

```

IMPORTANT

IF AN ERROR IS DETECTED ;IT COULD BE BECAUSE OF THREE REASONS.

- A) WRONG TEST NUMBER
- B) ERROR IN THE PRESENT TEST.
- C) POWER MONITOR BIT PROBLEM.

////////////////////////////////////
 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

ON POWER MONITOR ERRORS - FIND OUT WHAT POWER SUPPLY PROBLEM (OR CPU ERROR REGISTER PROBLEM) THERE IS AND OBTAIN REPAIR BEFORE RERUNNING THIS DIAGNOSTIC

159
160
161
162
163
164
165
166
167

7. RESTRICTIONS
7.1 STARTING RESTRICTION
NONE
7.2 OPERATIONAL RESTRICTION
NONE

168
169
170
171
172

8. MISCELLANEOUS

8.1 EXECUTION TIME

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

173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189

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.

190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218

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
		A	B
E-MODE/S-MODE (\$ENVM/\$ENV)		000/000	040/001
SWITCH REGISTER 1 (\$SWREG)		000000	000000
SWITCH REGISTER 2 CPU TYPE/OPTIONS		000000 00/0000	000000 00/0000

219
220
221
222
223
224
225
226
227

11.0

REVISION HISTORY

REVISION	DATE	COMMENT
CKKABA	MARCH 1979	ORIGINAL RELEASE
CKKABB	NOVEMBER 1980	FIX DIAGNOSTIC HALT WITH CIS SWITCH IN THE MAINTENANCE POSITION
CKKABC	APRIL 1981	INCLUSION OF BIT 0 CHECKING BEFORE EACH TEST.%

228
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333 000000
 334
 335 000000
 336 000001
 337 000001
 338 000002
 339 000003
 340 000003
 341 000004
 342 000005
 343 000005
 344 000006
 345 000006
 346 000007
 347 000007
 348 000000
 349 000003
 350 000004
 351 000014
 352 000030
 353 000020
 354 000034
 355 177564
 356 177560
 357 177564
 358 177566
 359 000240
 360 000240
 361 177776
 362 000010
 363 000010
 364 004700
 365 000100
 366 177776
 367 177766
 368 177413
 369 177776
 370

.TITLE CKKABCO 11/44 TRAPS
 :ALL INSTRUCTIONS THAT ARE RESERVED
 :SHOULD TRAP TO LOCATION 10, AND THE
 :PC THAT POINTS TO THE TRAPPING INSTRUCTION
 :SHOULD BE PLACED ON THE STACK
 :LISTING

.LIST ME
 .NLIST MC
 .NLIST MD
 .NLIST CND
 .ENABLE ABS
 .ENABLE AMA
 R0 =%0
 R1 =%1
 LAST =%1
 R2 =%2
 R3 =%3
 TAB =%3
 R4 =%4
 R5 =%5
 FIRST =%5
 R6 =%6
 SP =%6
 R7 =%7
 PC =%7
 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,.\$APTBL,.\$ACT11

378	000000	000002	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000002	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000004	000006	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000006	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000010	000012	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000012	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000014	000016	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000016	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000020	000022	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000022	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000024	000026	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000026	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000030	000032	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000032	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000034	000036	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000036	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000040	000042	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000042	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000044	000046	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000046	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000050	000052	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000052	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000054	000056	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000056	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000060	000062	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000062	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000064	000066	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000066	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000070	000072	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000072	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000074	000076	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000076	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000100	000102	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000102	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000104	000106	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000106	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000110	000112	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000112	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000114	000116	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000116	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000120	000122	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000122	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000124	000126	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000126	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000130	000132	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000132	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000134	000136	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000136	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000140	000142	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000142	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000144	000146	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000146	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000150	000152	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000152	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000154	000156	.WORD	.+2	:ADDRESS OF NEXT LOCATION
	000156	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
	000160	000162	.WORD	.+2	:ADDRESS OF NEXT LOCATION

000162	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000164	000166	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000166	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000170	000172	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000172	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000174	000176	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000176	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000200	000202	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000202	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000204	000206	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000206	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000210	000212	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000212	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000214	000216	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000216	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000220	000222	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000222	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000224	000226	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000226	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000230	000232	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000232	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000234	000236	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000236	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000240	000242	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000242	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000244	000246	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000246	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000250	000252	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000252	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000254	000256	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000256	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000260	000262	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000262	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000264	000266	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000266	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000270	000272	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000272	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000274	000276	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000276	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000300	000302	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000302	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000304	000306	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000306	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000310	000312	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000312	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000314	000316	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000316	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000320	000322	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000322	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000324	000326	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000326	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000330	000332	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000332	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000334	000336	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000336	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000340	000342	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000342	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS

000344	000346	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000346	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000350	000352	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000352	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000354	000356	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000356	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000360	000362	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000362	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000364	000366	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000366	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000370	000372	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000372	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS
000374	000376	.WORD	.+2	:ADDRESS OF NEXT LOCATION
000376	000000	HALT		:IF HALTED HERE, TRAPPED TO PREVIOUS ADDRESS


```

381      000200
382 000200 000137 000510
383      000210
384 000210 005037 000306
385 000214 000137 000510
386      000300
387

```

```

.=200
JMP      BEGIN
.=210
CLR      $PASS
JMP      BEGIN
.=300
.SBTTL   ACT11 HOOKS

```

```

:*****
:HOOKS REQUIRED BY ACT11

```

```

000046  000300
000052  000046
        023042
        000052
        000000
        000300

```

```

$SVPC=.
.=46
$ENDAD
.=52
.WORD   0
.=$SVPC

```

```

;SAVE PC
;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
;;2)SET LOC.52 TO ZERO
;; RESTORE PC

```


389

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

000330

```
.SBTTL APT MAILBOX-ETABLE
:*****
.EVEN
$MAIL:      ::APT MAILBOX
$MSGTY: .WORD  AMSGTY  ::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
$ETEND:
.MEXIT
```


391

000024 000330
000024 000024
000044 000200
000044 000044
000044 000330
000044 000330

000330
000330 000000
000332 000300
000334 000005
000336 000005
000340 000000
000342 000014

```
.SBTTL APT PARAMETER BLOCK
:*****
:SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
:*****
      .SX=.      ;;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
      =.SX     ;;RESET 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)
$STMT:  .WORD 5      ;;RUN TIM OF LONGEST TEST
$PASTM: .WORD 5      ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD 0      ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
      .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```


393 000304
394 000302
395
396 000500
397 000500 000000
398 000502 000000
399 000504 000250
400 000506 000252

\$TSTNM=\$TESTN
\$ERROR=\$FATAL
 .=500
BUFF: .WORD 0
RCPUER: .WORD 0
KTVEC: .WORD 250
KTSTA: .WORD 252

401	000510	012706	000500	BEGIN:	MOV	#500,SP	:SET UP STACK POINTER
402	000514	012737	177777		MOV	#-1,PASSPT	:CLEAR THE ITERATION COUNTER
403	000522	023737	000042		CMP	42,\$ENDAD	:SEE IF TITLE MESSAGE SHOULD BE PRINTED
404	000530	001403			BEQ	RESTRT	:BRANCH AROUND MESSAGE PRINTING IF SO
405	000532	004737	023510		JSR	PC,PRMSG	:GO PRINT MESSAGE USING
406	000536	023300			.WORD	TITLE	:STARTING ADDRESS OF THIS MESSAGE
407	000540	005037	000300	RESTRT:	CLR	\$MSGTY	
408	000544	012706	000500		MOV	#500,R6	
409	000550	012737	023424		MOV	#PWRDWN,24	:SET UP THE POWER DOWN VECTOR
410	000556	012737	000340		MOV	#340,26	:SET UP POWER DOWN PRIORITY
411	000564	012737	000006		MOV	#6,4	:SET UP TRAP VECTORS 4 & 6.
412	000572	005037	000006		CLR	6	
413	000576	012737	000012		MOV	#12,10	
414	000604	005037	000012		CLR	12	
415	000610	005037	000304		CLR	\$TSTNM	
416	000614	005037	000302		CLR	\$ERROR	
417	000620	012702	000300		MOV	#\$MSGTY,R2	
418	000624	000412			BR	TST1	

419
420 000000
421
422 000626 000000
423 000630 000000
424 000632 000000
425 000634 000000
426 000636 000000
427 000640 000000
428 000642 052525
429 000644 052400
430 000646 000000
431 000650 000000

;SPECIAL CASE OF ODD;.EVEN .BYTE AND REGISTER 6
HERE=0

K1: .WORD 0
K2: .WORD 0
K3: .WORD 0
K4: .WORD 0
K5: .WORD 0
K6: .WORD 0
K7: .WORD 052525
K10: .WORD 052400
K11: .WORD 0
K12: .WORD 0

432

```
.SBTTL TEST #1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
:*****
:TEST 1 - TEST AUTO INC AND DEC OF R6 FOR WORD AND BYTES
:*****
TST1: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;CPU POWER BIT FOUND SET
      BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #1,$TESTN ;SEQUENCE ERROR?
      BEQ 1000$ ;BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
      JMP TST2-12 ;JUMP TO ERROR HALT ON SEQ ERROR
1000$: CLR R6
      MOVB (R6)+,HERE ;SIX SHOULD INCREMENT BY TWO
      CMP R6,#2
      BEQ 1$
      MOV #1,$FATAL ;MOVE TO MAILBOX # ***** 1 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;R6 DID NOT AUTO INCREMENT BY TWO
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 764
```

000652 032737 000001 177766
 000660 001410
 000662 012737 000177 000302
 000670 005212
 000672 000000
 000674 042737 000001 177766
 000702 005237 000304
 000706 022737 000001 000304
 000714 001402
 000716 000137 001220
 000722
 433 000722 005006
 434 000724 112637 000000
 435 000730 020627 000002
 436 000734 001405
 000736 012737 000001 000302
 000744 005212
 000746 000000

437
 438 000750 012706 001000
 439 000754 114627 000000
 440 000760 020627 000776
 441 000764 001405
 000766 012737 000002 000302
 000774 005212
 000776 000000

442
 443 001000 005006
 444 001002 112626
 445 001004 020627 000004
 446 001010 001405
 001012 012737 000003 000302
 001020 005212
 001022 000000

447
 448 001024 005006
 449 001026 005004
 450 001030 122624
 451 001032 020627 000002
 452 001036 001405
 001040 012737 000004 000302
 001046 005212
 001050 000000

453
 454 001052 005006

```
1$: MOV #1000,R6
     MOVB -(R6),#HERE ;SHOULD DECREMENT BY TWO
     CMP R6,#776
     BEQ 2$
     MOV #2,$FATAL ;MOVE TO MAILBOX # ***** 2 *****
     INC (R2) ;SET MSGTYP TO FATAL ERROR
     HALT ;R6 DID NOT AUTO DECREMENT BY 2
     ;TO SCOPE REPLACE HALT WITH 240
     ;AND REPLACE NEXT INST WITH 750

2$: CLR R6
     MOVB (R6)+,(R6)+ ;DOUBLES AUTO INCREMENT OF R6
     CMP R6,#4
     BEQ 3$
     MOV #3,$FATAL ;MOVE TO MAILBOX # ***** 3 *****
     INC (R2) ;SET MSGTYP TO FATAL ERROR
     HALT ;WRONG AUTO INCREMENT OF R6
     ;TO SCOPE REPLACE HALT WITH 240
     ;AND REPLACE NEXT INST WITH 736

3$: CLR R6
     CLR R4
     CMPB (R6)+,(R4)+ ;TEST INCREMENT OF R6
     CMP R6,#2
     BEQ 4$
     MOV #4,$FATAL ;MOVE TO MAILBOX # ***** 4 *****
     INC (R2) ;SET MSGTYP TO FATAL ERROR
     HALT ;WRONG INCREMENT OF R6
     ;TO SCOPE REPLACE HALT WITH 240
     ;AND REPLACE NEXT INST WITH 723

4$: CLR R6
```


482

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

 :TEST 2 - TEST TRANSFER OF .BYTE USING R6

001232	032737	00000.	177766	TST2:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001240	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
001242	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
001250	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001252	000000				HALT		:CPU POWER BIT FOUND SET
001254	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
001262	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
001266	022737	000002	000304		CMP	#2,\$TESTN	:SEQUENCE ERROR?
001274	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001276	000137	001600			JMP	TST3-12	:JUMP TO ERROR HALT ON SEQ ERROR
001302				1000\$:			
483	001302	012737	123456		MOV	#123456,K5	
484	001310	012737	050505		MOV	#050505,K1	
485	001316	012705	000626		MOV	#K1,R5	:R5=(050505)K1
486	001322	012706	000636		MOV	#K5,R6	:R6=(123456)K5
487	001326	112625			MOVB	(R6)+,(R5)+	:LOW .BYTE OF R6 TO R5
488	001330	022737	050456		CMP	#050456,K1	
489	001336	001405			BEQ	1\$	
	001340	012737	000012		MOV	#12,\$FATAL	:MOVE TO MAILBOX # ***** 12 *****
	001346	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	001350	000000			HALT		:FALSE TRANSFER OF .BYTE
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753

490

491	001352	012737	123456		1\$:	MOV	#123456,K5	
492	001360	012737	050505			MOV	#050505,K1	
493	001366	012705	000626			MOV	#K1,R5	:R5(050505)K1
494	001372	012706	000640			MOV	#K6,R6	:R6(123456)K5
495	001376	114625				MOVB	-(R6),(R5)+	:LOW .BYTE OF R6 TO R5 (DECREMENT)
496	001400	023727	000626	050456		CMP	K1,#050456	
497	001406	001405				BEQ	2\$	
	001410	012737	000013	000302		MOV	#13,\$FATAL	:MOVE TO MAILBOX # ***** 13 *****
	001416	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
	001420	000000				HALT		:FALSE R6 .BYTE TRANSFER
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 727

498

499	001422	012737	123456		2\$:	MOV	#123456,K1	
500	001430	012737	050505			MOV	#050505,K5	
501	001436	012705	000626			MOV	#K1,R5	:(123456)
502	001442	012706	000636			MOV	#K5,R6	:(050505)
503	001446	112526				MOVB	(R5)+,(R6)+	:LOW OF R5 TO LOW OF R6
504	001450	022737	050456			CMP	#050456,K5	
505	001456	001405				BEQ	3\$	
	001460	012737	000014	000302		MOV	#14,\$FATAL	:MOVE TO MAILBOX # ***** 14 *****
	001466	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
	001470	000000				HALT		:FALSE R6 .BYTE TRANSFER
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 703

506

507	001472	012737	123456		3\$:	MOV	#123456,K1	
508	001500	012737	050505			MOV	#050505,K5	
509	001506	012705	000627			MOV	#K1+1,R5	:123456

510	001512	012706	000636		MOV	#K5,R6	:050505
511	001516	112526			MOVB	(R5)+,(R6)+	:HIGH OF R5 TO LOW OF R6
512	001520	023727	000636	050647	CMP	K5,#050647	
513	001526	001405			BEQ	4\$	
	001530	012737	000015	000302	MOV	#15,\$FATAL	:MOVE TO MAILBOX # ***** 15 *****
	001536	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	001540	000000			HALT		:FALSE R6 .BYTE TRANSFER
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 657
514							
515	001542	012737	123456	000626	4\$:	MOV	#123456,K1
516	001550	012737	050505	000636		MOV	#050505,K5
517	001556	012705	000627			MOV	#K1+1,R5
518	001562	012706	000636			MOV	#K5,R6
519	001566	112625			MOVB	(R6)+,(R5)+	:R5-123456-ODD ADDRESS
520	001570	022737	042456	000626	CMP	#042456,K1	:R6-050505--.EVEN ADDRESS
521	001576	001405			BEQ	TST3	:LOW OF R6 TO HIGH OF R5
	001600	012737	000016	000302	MOV	#16,\$FATAL	:MOVE TO MAILBOX # ***** 16 *****
	001606	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	001610	000000			HALT		:FAILED LOW OF 6 TO HIGH OF 5,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 633

522

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

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

001612	032737	000001	177766	TST3:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
001620	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
001622	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
001630	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001632	000000				HALT		:CPU POWER BIT FOUND SET
001634	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
001642	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
001646	022737	000003	000304		CMP	#3,\$TESTN	:SEQUENCE ERROR?
001654	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
001656	000137	002070			JMP	TST4-12	:JUMP TO ERROR HALT ON SEQ ERROR
001662				1000\$:			
523 001662	123737	000642	000643		CMPB	K7,K7+1	:SAME .WORD LOW TO HIGH
524 001670	001405				BEQ	5\$	
001672	012737	000017	000302		MOV	#17,\$FATAL	:MOVE TO MAILBOX # ***** 17 *****
001700	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001702	000000				HALT		:SHOULD COMPARE LOW TO HIGH
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 766
525 001704	123737	000643	000642	5\$:	CMPB	K7+1,K7	:COMPARE ODD TO .EVEN SAME .WORD
526 001712	001405				BEQ	6\$	
527 001714	012737	000020	000302		MOV	#20,\$FATAL	:MOVE TO MAILBOX # ***** 20 *****
001722	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001724	000000				HALT		:ODD TO .EVEN .BYTE FAILURE
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
528 001726	123737	000645	000642	6\$:	CMPB	K10+1,K7	:SEQUENTIAL .BYTES
529 001734	001405				BEQ	7\$	
530 001736	012737	000021	000302		MOV	#21,\$FATAL	:MOVE TO MAILBOX # ***** 21 *****
001744	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001746	000000				HALT		:ODD TO .EVEN FAILED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 744
531 001750	123737	000644	000640	7\$:	CMPB	K10,K6	
532 001756	001405				BEQ	8\$	
533 001760	012737	000022	000302		MOV	#22,\$FATAL	:MOVE TO MAILBOX # ***** 22 *****
001766	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
001770	000000				HALT		:.EVEN TO EVEN FAILED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 733
534 001772	123737	000643	000645	8\$:	CMPB	K7+1,K10+1	
535 002000	001405				BEQ	9\$	
002002	012737	000023	000302		MOV	#23,\$FATAL	:MOVE TO MAILBOX # ***** 23 *****
002010	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002012	000000				HALT		:ODD TO ODD FAILED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 722
536 002014	123737	000644	000645	9\$:	CMPB	K10,K10+1	
537 002022	001005				BNE	10\$	
538 002024	012737	000024	000302		MOV	#24,\$FATAL	:MOVE TO MAILBOX # ***** 24 *****

	002032	005212				INC	(R2)		:SET MSGTYP TO FATAL ERROR
	002034	000000				HALT			:LOW TO HIGH IN SAME .WORD FAILED
									:TO SCOPE REPLACE HALT WITH 240
									:AND REPLACE NEXT INST WITH 711
539									
540	002036	123737	000645	000645	10\$:	CMPB	K10+1,K10+1		
541	002044	001405				BEQ	11\$		
	002046	012737	000025	000302		MOV	#25,\$FATAL		:MOVE TO MAILBOX # ***** 25 *****
	002054	005212				INC	(R2)		:SET MSGTYP TO FATAL ERROR
	002056	000000				HALT			:HIGH TO LOW IN SAME .WORD FAILED
									:TO SCOPE REPLACE HALT WITH 240
									:AND REPLACE NEXT INST WITH 700
542									
543	002060	123737	000644	000643	11\$:	CMPB	K10,K7+1		
544	002066	001005				BNE	TST4		
	002070	012737	000026	000302		MOV	#26,\$FATAL		:MOVE TO MAILBOX # ***** 26 *****
	002076	005212				INC	(R2)		:SET MSGTYP TO FATAL ERROR
	002100	000000				HALT			:EVEN TO ODD FAILED,OR WRONG \$STNM
									:TO SCOPE REPLACE HALT WITH 240
									:AND REPLACE NEXT INST WITH 667

545

.SBTTL TEST #4 - TEST THE CC BITS

 :TEST 4 - TEST THE CC BITS

002102	032737	000001	177766	TST4:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002110	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
002112	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
002120	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002122	000000				HALT		:CPU POWER BIT FOUND SET
002124	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
002132	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
002136	022737	000004	000304		CMP	#4,\$TESTN	:SEQUENCE ERROR?
002144	001062				BNE	TST5-12	:BRANCH TO ERROR HALT ON SEQ ERROR
546 002146	000277				SCC		:SET STATUS
547 002150	005037	177776			CLR	STATUS	:CLEAR STATUS
548 002154	103005				BCC	1\$	
002156	012737	000027	000302		MOV	#27,\$FATAL	:MOVE TO MAILBOX # ***** 27 *****
002164	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002166	000000				HALT		:C NOT CLEAR :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 766
549 002170				1\$:			
002170	102005				BVC	2\$	
002172	012737	000030	000302		MOV	#30,\$FATAL	:MOVE TO MAILBOX # ***** 30 *****
002200	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002202	000000				HALT		:V NOT CLEAR :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 760
550 002204				2\$:			
002204	001005				BNE	3\$	
002206	012737	000031	000302		MOV	#31,\$FATAL	:MOVE TO MAILBOX # ***** 31 *****
002214	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002216	000000				HALT		:Z NOT CLEAR :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 752
551 002220				3\$:			
002220	100005				BPL	4\$	
002222	012737	000032	000302		MOV	#32,\$FATAL	:MOVE TO MAILBOX # ***** 32 *****
002230	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002232	000000				HALT		:N NOT CLEAR :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 744
552 002234	000257			4\$:			
553 002236	052737	000017	177776		CCC		:CLEAR CONDITION CODES
554					BIS	#17,STATUS	:SET STATUS TO ONES
555 002244	103405				BCS	5\$	
002246	012737	000033	000302		MOV	#33,\$FATAL	:MOVE TO MAILBOX # ***** 33 *****
002254	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002256	000000				HALT		:C NOT SET :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 732
556 002260				5\$:			
002260	102405				BVS	6\$	
002262	012737	000034	000302		MOV	#34,\$FATAL	:MOVE TO MAILBOX # ***** 34 *****
002270	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002272	000000				HALT		:V NOT SET :TO SCOPE REPLACE HALT WITH 240


```
557 002274      -      6$:      BEQ      7$
      002274 001405      MOV      #35,$FATAL
      002276 012737 000035 000302  INC      (R2)
      002304 005212      HALT
      002306 000000

;AND REPLACE NEXT INST WITH 724
;MOVE TO MAILBOX # ***** 35 *****
;SET MSGTYP TO FATAL ERROR
;Z NOT SET
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 716

558 002310      -      7$:      BMI      TST5
      002310 100405      MOV      #36,$FATAL
      002312 012737 000036 000302  INC      (R2)
      002320 005212      HALT
      002322 000000

;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
```


559

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

 :TEST 5 - TEST THAT A TRAP OCCURS ON A RESERVED INS

002324	032737	000001	177766	TST5:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002332	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
002334	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
002342	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002344	000000				HALT		:CPU POWER BIT FOUND SET
002346	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
002354	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
002360	022737	000005	000304		CMP	#5,\$TESTN	:SEQUENCE ERROR?
002366	001006				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
560 002370	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
561 002374	012737	002416	000010		MOV	#2\$,RTRAP	:RETURN LOCATION
562 002402	000010				TRAPA		:RESERVED INSTRUCTION, SHOULD TRAP
563 002404				1\$:			
002404	012737	000037	000302		MOV	#37,\$FATAL	:MOVE TO MAILBOX # ***** 37 *****
002412	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002414	000000				HALT		:RESERVE INSTRUCTION DIDN'T TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 764
564 002416				2\$:			

565

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

 :TEST 6 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

002416	032737	000001	177766	TST6:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002424	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
002426	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
002434	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002436	000000				HALT		:CPU POWER BIT FOUND SET
002440	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
002446	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
002452	022737	000006	000304		CMP	#6,\$TESTN	:SEQUENCE ERROR?
002460	001011				BNE	TST7-12	:BRANCH TO ERROR HALT ON SEQ ERROR
566 002462	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
567 002466	012737	002476	000010		MOV	#1\$,RTRAP	:RETURN POINTER
568 002474	000010				TRAPA		:RESERVED INSTRUCTION
569 002476	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
570 002502	001405				BEQ	TST7	
002504	012737	000040	000302		MOV	#40,\$FATAL	:MOVE TO MAILBOX # ***** 40 *****
002512	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002514	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

571

.SBTTL TEST #7 - TEST THAT PROPER P.C. IS SAVED

:TEST 7 - TEST THAT PROPER P.C. IS SAVED

002516	032737	000001	177766	TST7:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
002524	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
002526	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
002534	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002536	000000				HALT		:CPU POWER BIT FOUND SET
002540	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
002546	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
002552	022737	000007	000304		CMP	#7,\$TESTN	:SEQUENCE ERROR?
002560	001012				BNE	TST10-12	:BRANCH TO ERROR HALT ON SEQ ERROR
572 002562	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
573 002566	012737	002576	000010		MOV	#1\$,RTRAP	:RETURN FROM TRAP POINTER
574 002574	000010				TRAPA		:TRAP ON THIS INSTRUCTION
575 002576	022737	002576	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK FOR INCREMENTED P.C.
576 002604	001405				BEQ	TST10	
002606	012737	000041	000302		MOV	#41,\$FATAL	:MOVE TO MAILBOX # ***** 41 *****
002614	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
002616	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

577

```

.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: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
        BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
        MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;CPU POWER BIT FOUND SET
        BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #10,$TESTN ;SEQUENCE ERROR?
      BNE TST11-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
578 002664 012706 000500 MOV #BUFF,SP ;SET UP
579 002670 012737 002706 000010 MOV #1$,RTRAP ;SET UP
580 002676 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
581 002702 000257 CCC
582 002704 000010 TRAPA ;TRAP
583 002706 023727 000476 000000 1$: CMP BUFF-2,#0 ;TEST THAT OLD STATUS WENT TO STACK
      BEQ 2$
      MOV #42,$FATAL ;MOVE TO MAILBOX # ***** 42 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT STATUS
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 755
585 002730 012706 000500 2$: MOV #BUFF,SP ;SET UP
586 002734 012737 002754 000010 MOV #3$,RTRAP ;SET UP
587 002742 012737 000357 177776 MOV #357,CC ;SET PRIORITY
588 002750 000277 SCC ;SET CC
589 002752 000010 TRAPA ;TRAP
590 002754 023727 000476 000357 3$: 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 $STNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 732
  
```


592

.SBTTL TEST #11 - TEST THAT 'NEW' STATUS IS CORRECT

 :TEST 11 - TEST THAT 'NEW' STATUS IS CORRECT

002776	032737	000001	177766	TST11:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003004	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
003006	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
003014	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003016	000000				HALT		:CPU POWER BIT FOUND SET
003020	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
003026	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
003032	022737	000011	000304		CMP	#11,\$TESTN	:SEQUENCE ERROR?
003040	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
003042	000137	003310			JMP	12\$:JUMP TO ERROR HALT ON SEQ ERROR
003046				1000\$:			
593 003046	012706	000500			MOV	#BUFF,SP	
594 003052	012737	003066	000010		MOV	#1\$,RTRAP	
595 003060	005037	000012			CLR	RTRAP+2	:CLEAR FUTURE PRIORITY AND CC
596 003064	000010				TRAPA		
597 003066				1\$:			:TEST FOR 'C' CLEARED
598 003066	100005				BPL	2\$	
003070	012737	000044	000302		MOV	#44,\$FATAL	:MOVE TO MAILBOX # ***** 44 *****
003076	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003100	000000				HALT		:N NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 761
599 003102				2\$:			
003102	001005				BNE	3\$	
003104	012737	000045	000302		MOV	#45,\$FATAL	:MOVE TO MAILBOX # ***** 45 *****
003112	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003114	000000				HALT		:Z NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 753
600 003116				3\$:			
003116	102005				BVC	4\$	
003120	012737	000046	000302		MOV	#46,\$FATAL	:MOVE TO MAILBOX # ***** 46 *****
003126	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003130	000000				HALT		:V NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 745
601 003132				4\$:			
003132	103005				BCC	5\$	
003134	012737	000047	000302		MOV	#47,\$FATAL	:MOVE TO MAILBOX # ***** 47 *****
003142	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003144	000000				HALT		:C NOT CLEARED :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 737
602 003146	032737	000340	177776	5\$:	BIT	#340,CC	:TEST PRIORITY
603 003154	001405				BEQ	6\$	
003156	012737	000050	000302		MOV	#50,\$FATAL	:MOVE TO MAILBOX # ***** 50 *****
003164	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003166	000000				HALT		:PRIORITY NOT ZERO :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 726
604 003170	012706	000500		6\$:	MOV	#BUFF,SP	
605 003174	012737	003212	000010		MOV	#7\$,RTRAP	
606 003202	012737	000357	000012		MOV	#357,RTRAP+2	:SET NEW 'CC' AND PRIORITY

607	003210	000010			TRAPA					:TRAP HERE
608	003212				7\$: BMI	8\$				
	003212	100405				#51,\$FATAL				:MOVE TO MAILBOX # ***** 51 *****
	003214	012737	000051	000302		MOV				:SET MSGTYP TO FATAL ERROR
	003222	005212				INC	(R2)			:N NOT SET
	003224	000000				HALT				:TO SCOPE REPLACE HALT WITH 240
										:AND REPLACE NEXT INST WITH 707
609	003226				8\$: BEQ	9\$				
	003226	001405				#52,\$FATAL				:MOVE TO MAILBOX # ***** 52 *****
	003230	012737	000052	000302		MOV				:SET MSGTYP TO FATAL ERROR
	003236	005212				INC	(R2)			:Z NOT SET
	003240	000000				HALT				:TO SCOPE REPLACE HALT WITH 240
										:AND REPLACE NEXT INST WITH 701
610	003242				9\$: BVS	10\$				
	003242	102405				#53,\$FATAL				:MOVE TO MAILBOX # ***** 53 *****
	003244	012737	000053	000302		MOV				:SET MSGTYP TO FATAL ERROR
	003252	005212				INC	(R2)			:V NOT SET
	003254	000000				HALT				:TO SCOPE REPLACE HALT WITH 240
										:AND REPLACE NEXT INST WITH 673
611	003256				10\$: BCS	11\$				
	003256	103405				#54,\$FATAL				:MOVE TO MAILBOX # ***** 54 *****
	003260	012737	000054	000302		MOV				:SET MSGTYP TO FATAL ERROR
	003266	005212				INC	(R2)			:C NOT SET
	003270	000000				HALT				:TO SCOPE REPLACE HALT WITH 240
										:AND REPLACE NEXT INST WITH 665
612	003272	013706	177776		11\$: MOV	CC,SP				
613	003276	042706	000017			#17,SP				
614	003302	022706	000340			#340,SP				
615	003306	001405				13\$				
	003310				12\$: MOV	#55,\$FATAL				:MOVE TO MAILBOX # ***** 55 *****
	003310	012737	000055	000302		INC	(R2)			:SET MSGTYP TO FATAL ERROR
	003316	005212				HALT				:PRIORITY WAS CHANGED,OR WRONG \$STNM
	003320	000000								:TO SCOPE REPLACE HALT WITH 240
										:AND REPLACE NEXT INST WITH 651
616	003322	012737	000012	000010	13\$: MOV	#12,10				
617	003330	005037	000012			CLR	12			

618

.SBTTL TEST #12 - TEST THAT A TRAP OCCURS FOR A 'TRAP' INSTRUCTION
 :*****
 :TEST 12 - TEST THAT A TRAP OCCURS FOR A 'TRAP' INSTRUCTION
 :*****

003334	032737	000001	177766	TST12:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003342	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
003344	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
003352	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003354	000000				HALT		:CPU POWER BIT FOUND SET
003356	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
003364	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
003370	022737	000012	000304		CMP	#12,\$TESTN	:SEQUENCE ERROR?
003376	001013				BNE	TST13-12	:BRANCH TO ERROR HALT ON SEQ ERROR
619 003400	012737	000012	000010		MOV	#12,10	
620 003406	005037	000012			CLR	12	
621 003412	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
622 003416	012737	003440	000034		MOV	#1\$,RTRAP1	:RETURN LOCATION
623 003424	104400				TRAP		:RESERVED INSTRUCTION, SHOULD TRAP
624 003426	012737	000056	000302		MOV	#56,\$FATAL	:MOVE TO MAILBOX # ***** 56 *****
	003434	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	003436	000000			HALT		:TRAP DIDN'T TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 757

625 003440

1\$:

626

.SBTTL TEST #13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

 :TEST 13 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

003440	032737	000001	177766	TST13:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003446	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
003450	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
003456	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003460	000000				HALT		:CPU POWER BIT FOUND SET
003462	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
003470	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
003474	022737	000013	000304		CMP	#13,\$TESTN	:SEQUENCE ERROR?
003502	001011				BNE	TST14-12	:BRANCH TO ERROR HALT ON SEQ ERROR
627 003504	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
628 003510	012737	003520	000034		MOV	#1\$,RTRAP1	:RETURN POINTER
629 003516	104400				TRAP		:RESERVED INSTRUCTION
630 003520	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
631 003524	001405				BEQ	TST14	
003526	012737	000057	000302		MOV	#57,\$FATAL	:MOVE TO MAILBOX # ***** 57 *****
003534	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003536	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM :TO SCOPE REPLACE HALT WITH 240 :AND REPLACE NEXT INST WITH 761

632

.SBTTL TEST #14 - TEST THAT PROPER P.C. IS SAVED

:TEST 14 - TEST THAT PROPER P.C. IS SAVED

003540	032737	000001	177766	TST14:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003546	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
003550	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
003556	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003560	000000				HALT		:CPU POWER BIT FOUND SET
003562	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
003570	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
003574	022737	000014	000304		CMP	#14,\$TESTN	:SEQUENCE ERROR?
003602	001012				BNE	TST15-12	:BRANCH TO ERROR HALT ON SEQ ERROR
633 003604	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
634 003610	012737	003620	000034		MOV	#1\$,RTRAP1	:RETURN FROM TRAP POINTER
635 003616	104400				TRAP		:TRAP ON THIS INSTRUCTION
636 003620	022737	003620	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK INCREMENTED P.C.
637 003626	001405				BEQ	TST15	
003630	012737	000060	000302		MOV	#60,\$FATAL	:MOVE TO MAILBOX # ***** 60 *****
003636	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003640	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

638

.SBTTL TEST #15 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK

 :TEST 15 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK

003642	032737	000001	177766	TST15:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
003650	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
003652	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
003660	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003662	000000				HALT		:CPU POWER BIT FOUND SET
003664	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
003672	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
003676	022737	000015	000304		CMP	#15,\$TESTN	:SEQUENCE ERROR?
003704	001037				BNE	TST16-12	:BRANCH TO ERROR HALT ON SEQ ERROR
639 003706	012706	000500			MOV	#BUFF,SP	:SET UP
640 003712	012737	003730	000034		MOV	#1\$,RTRAP1	:SET UP
641 003720	005037	177776			CLR	CC	:CLEAR CC AND PRIORITY
642 003724	000257				CCC		
643 003726	104400				TRAP		:TRAP
644 003730	023727	000476	000000	1\$:	CMP	BUFF-2,#0	:TEST THAT OLD STATUS WENT TO STACK
645 003736	001405				BEQ	2\$	
003740	012737	000061	000302		MOV	#61,\$FATAL	:MOVE TO MAILBOX # ***** 61 *****
003746	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
003750	000000				HALT		:INCORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
646 003752	012706	000500		2\$:	MOV	#BUFF,SP	:SET UP
647 003756	012737	003774	000034		MOV	#3\$,RTRAP1	:SET UP
648 003764	012737	000357	177776		MOV	#357,CC	:SET PRIORITY
649 003772	104400				TRAP		:SET CC
650 003774	023727	000476	000357	3\$:	CMP	BUFF-2,#357	:COMPARES STATUS ON STACK
651 004002	001405				BEQ	TST16	
004004	012737	000062	000302		MOV	#62,\$FATAL	:MOVE TO MAILBOX # ***** 62 *****
004012	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004014	000000				HALT		:INCORRECT STATUS ON STACK,OR WRONG \$TSTNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 733

652

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

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

004016	032737	000001	177766	TST16:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004024	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
004026	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
004034	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004036	000000				HALT		:CPU POWER BIT FOUND SET
004040	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
004046	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
004052	022737	000016	000304		CMP	#16,\$TESTN	:SEQUENCE ERROR?
004060	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
004062	000137	004330			JMP	TST17-12	:JUMP TO ERROR HALT ON SEQ ERROR
004066				1000\$:			
653 004066	012706	000500			MOV	#BUFF,SP	
654 004072	012737	004106	000034		MOV	#1\$,RTRAP1	
655 004100	005037	000036			CLR	RTRAP1+2	:CLEAR FUTURE PRIORITY AND CC
656 004104	104400				TRAP		
657 004106				1\$:			:TEST FOR 'C' CLEARED
658 004106	100005				BPL	2\$	
004110	012737	000063	000302		MOV	#63,\$FATAL	:MOVE TO MAILBOX # ***** 63 *****
004116	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004120	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
659 004122				2\$:			
004122	001005				BNE	3\$	
004124	012737	000064	000302		MOV	#64,\$FATAL	:MOVE TO MAILBOX # ***** 64 *****
004132	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004134	000000				HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
660 004136				3\$:			
004136	102005				BVC	4\$	
004140	012737	000065	000302		MOV	#65,\$FATAL	:MOVE TO MAILBOX # ***** 65 *****
004146	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004150	000000				HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
661 004152				4\$:			
004152	103005				BCC	5\$	
004154	012737	000066	000302		MOV	#66,\$FATAL	:MOVE TO MAILBOX # ***** 66 *****
004162	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004164	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
662 004166	032737	000340	177776	5\$:	BIT	#340,CC	:TEST PRIORITY
663 004174	001405				BEQ	6\$	
004176	012737	000067	000302		MOV	#67,\$FATAL	:MOVE TO MAILBOX # ***** 67 *****
004204	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004206	000000				HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
664 004210	012706	000500		6\$:	MOV	#BUFF,SP	
665 004214	012737	004232	000034		MOV	#7\$,RTRAP1	
666 004222	012737	000357	000036		MOV	#357,RTRAP1+2	:SET NEW 'CC' AND PRIORITY


```

667 004230 104400 TRAP ;TRAP HERE
668 004232 7$: BMI 8$
669 004232 100405 000070 000302 MOV #70,$FATAL ;MOVE TO MAILBOX # ***** 70 *****
    004234 012737 INC (R2) ;SET MSGTYP TO FATAL ERROR
    004242 005212 HALT ;N NOT SET
    004244 000000 ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 707

670 004246 8$: BEQ 9$
    004246 001405 000071 000302 MOV #71,$FATAL ;MOVE TO MAILBOX # ***** 71 *****
    004250 012737 INC (R2) ;SET MSGTYP TO FATAL ERROR
    004256 005212 HALT ;Z NOT SET
    004260 000000 ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 701

671 004262 9$: BVS 10$
    004262 102405 000072 000302 MOV #72,$FATAL ;MOVE TO MAILBOX # ***** 72 *****
    004264 012737 INC (R2) ;SET MSGTYP TO FATAL ERROR
    004272 005212 HALT ;V NOT SET
    004274 000000 ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 673

672 004276 10$: BCS 11$
    004276 103405 000073 000302 MOV #73,$FATAL ;MOVE TO MAILBOX # ***** 73 *****
    004300 012737 INC (R2) ;SET MSGTYP TO FATAL ERROR
    004306 005212 HALT ;C NOT SET
    004310 000000 ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 665

673 004312 11$: MOV CC,SP
674 004316 042706 000017 BIC #17,SP
675 004322 022706 000340 CMP #340,SP
676 004326 001405 BEQ TST17
    004330 012737 000074 000302 MOV #74,$FATAL ;MOVE TO MAILBOX # ***** 74 *****
    004336 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    004340 000000 HALT ;PRIORITY WAS CHANGED,OR WRONG $STNM
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 651
    
```


677

.SBTTL TEST #17 - TEST THAT ALL COMB 'TRAP' WILL CAUSE A TRAP

 :TEST 17 - TEST THAT ALL COMB 'TRAP' WILL CAUSE A TRAP

004342	032737	000001	177766	TST17:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004350	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
004352	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
004360	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004362	000000				HALT		:CPU POWER BIT FOUND SET
004364	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
004372	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
004376	022737	000017	000304		CMP	#17,\$TESTN	:SEQUENCE ERROR?
004404	001011				BNE	3\$:BRANCH TO ERROR HALT ON SEQ ERROR
678 004406	012737	104400	004426		MOV	#TRAP,2\$:INITIALIZE BASE TRAP INSTRUCTION
679 004414	012737	004442	000034		MOV	#4\$,34	:RETURN FROM TRAP TO RA1
680 004422	012706	000500		1\$:	MOV	#BUFF,SP	:SET UP STACK POINTER
681 004426	104400			2\$:	TRAP		:TRAP INST WILL BE MODIFIED TO TRAP+377
682 004430				3\$:			
004430	012737	000075	000302		MOV	#75,\$FATAL	:MOVE TO MAILBOX # ***** 75 *****
004436	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004440	000000				HALT		:PREVIOUS INST FAILED TO TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
683 004442	005237	004426		4\$:	INC	2\$:INCREMENT TRAP INSTRUCTION
684 004446	022737	104777	004426		CMP	#104777,2\$:TRAP+377 TO UPPER LIMIT
685 004454	103362				BHIS	1\$:HAVE WE TESTED ALL
686 004456	012737	000036	000034		MOV	#36,34	
687 004464	005037	000036			CLR	36	

688

```
.SBTTL TEST #20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
:*****
:TEST 20 - TEST THAT A TRAP OCCURES ON AN "IOT" INSTRUCTION
:*****
TST20: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;CPU POWER BIT FOUND SET
      BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #20,$TESTN ;SEQUENCE ERROR?
      BNE TST21-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
      MOV #BUFF,SP ;STACK POINTER SETUP
      MOV #1$,RTRAP2 ;RETURN LOCATION
      IOT ;RESERVE INSTRUCTION, SHOULD TRAP
      MOV #76,$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
```

```
004470 032737 000001 177766
004476 001410
004500 012737 000177 000302
004506 005212
004510 000000
004512 042737 000001 177766
004520 005237 000304
004524 022737 000020 000304
004532 001006
689 004534 012706 000500
690 004540 012737 004562 000020
691 004546 000004
692 004550 012737 000076 000302
    004556 005212
    004560 000000
```

693 004562

1\$:

694

.SBTTL TEST #21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

 :TEST 21 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

004562	032737	000001	177766	TST21:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004570	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
004572	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
004600	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004602	000000				HALT		:CPU POWER BIT FOUND SET
004604	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
004612	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
004616	022737	000021	000304		CMP	#21,\$TESTN	:SEQUENCE ERROR?
004624	001011				BNE	TST22-12	:BRANCH TO ERROR HALT ON SEQ ERROR
695 004626	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
696 004632	012737	004642	000020		MOV	#1\$,RTRAP2	:RETURN POINTER
697 004640	000004				IOT		:RESERVED INSTRUCTION
698 004642	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
699 004646	001405				BEQ	TST22	
004650	012737	000077	000302		MOV	#77,\$FATAL	:MOVE TO MAILBOX # ***** 77 *****
004656	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004660	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

700

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

:TEST 22 - TEST THAT PROPER P.C. IS SAVED

004662	032737	000001	177766	TST22:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004670	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
004672	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
004700	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004702	000000				HALT		:CPU POWER BIT FOUND SET
004704	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
004712	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
004716	022737	000022	000304		CMP	#22,\$TESTN	:SEQUENCE ERROR?
004724	001012				BNE	TST23-12	:BRANCH TO ERROR HALT ON SEQ ERROR
701 004726	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
702 004732	012737	004742	000020		MOV	#1\$,RTRAP2	:RETURN FROM TRAP POINTER
703 004740	000004				IOT		:TRAP ON THIS INSTRUCTION
704 004742	022737	004742	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK FOR INCREMENTED P.C.
705 004750	001405				BEQ	TST23	
004752	012737	000100	000302		MOV	#100,\$FATAL	:MOVE TO MAILBOX # ***** 100 *****
004760	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
004762	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

706

.SBTTL TEST #23 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK

 :TEST 23 - TEST THAT 'OLD' CC AND PRI ARE PLACED ON STACK

004764	032737	000001	177766	TST23:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
004772	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
004774	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
005002	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005004	000000				HALT		:CPU POWER BIT FOUND SET
005006	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
005014	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
005020	022737	000023	000304		CMP	#23,\$TESTN	:SEQUENCE ERROR?
005026	001040				BNE	TST24-12	:BRANCH TO ERROR HALT ON SEQ ERROR
707 005030	012706	000500			MOV	#BUFF,SP	:SET UP
708 005034	012737	005052	000020		MOV	#1\$,RTRAP2	:SET UP
709 005042	005037	177776			CLR	CC	:CLEAR CC AND PRIORITY
710 005046	000257				CCC		
711 005050	000004				IOT		:TRAP
712 005052	023727	000476	000000	1\$:	CMP	BUFF-2,#0	:TEST THAT OLD STATUS WENT TO STACK
713 005060	001405				BEQ	2\$	
005062	012737	000101	000302		MOV	#101,\$FATAL	:MOVE TO MAILBOX # ***** 101 *****
005070	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005072	000000				HALT		:INCORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
714 005074	012706	000500		2\$:	MOV	#BUFF,SP	:SET UP
715 005100	012737	005120	000020		MOV	#3\$,RTRAP2	:SET UP
716 005106	012737	000357	177776		MOV	#357,CC	:SET PRIORITY
717 005114	000277				SCC		:SET CC
718 005116	000004				IOT		:TRAP
719 005120	023727	000476	000357	3\$:	CMP	BUFF-2,#357	:COMPARES STATUS ON STACK
720 005126	001405				BEQ	TST24	
005130	012737	000102	000302		MOV	#102,\$FATAL	:MOVE TO MAILBOX # ***** 102 *****
005136	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005140	000000				HALT		:INCORRECT STATUS ON STACK,OR WRONG \$1STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 732

721

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

 :TEST 24 - TEST THAT 'NEW' STATUS IS CORRECT

005142	032737	000001	177766	TST24:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005150	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
005152	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
005160	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005162	000000				HALT		:CPU POWER BIT FOUND SET
005164	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
005172	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
005176	022737	000024	000304		CMP	#24,\$TESTN	:SEQUENCE ERROR?
005204	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
005206	000137	005454			JMP	12\$:JUMP TO ERROR HALT ON SEQ ERROR
005212				1000\$:			
722	005212	012706	000500		MOV	#BUFF,SP	
723	005216	012737	005232	000020	MOV	#1\$,RTRAP2	
724	005224	005037	000022		CLR	RTRAP2+2	:CLEAR FUTURE PRIORITY AND CC
725	005230	000004			IOT		
726	005232			1\$:			:TEST FOR 'C' CLEARED
727	005232	100005			BPL	2\$	
	005234	012737	000103	000302	MOV	#103,\$FATAL	:MOVE TO MAILBOX # ***** 103 *****
	005242	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	005244	000000			HALT		:N NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
728	005246			2\$:			
	005246	001005			BNE	3\$	
	005250	012737	000104	000302	MOV	#104,\$FATAL	:MOVE TO MAILBOX # ***** 104 *****
	005256	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	005260	000000			HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
729	005262			3\$:			
	005262	102005			BVC	4\$	
	005264	012737	000105	000302	MOV	#105,\$FATAL	:MOVE TO MAILBOX # ***** 105 *****
	005272	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	005274	000000			HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
730	005276			4\$:			
	005276	103005			BCC	5\$	
	005300	012737	000106	000302	MOV	#106,\$FATAL	:MOVE TO MAILBOX # ***** 106 *****
	005306	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	005310	000000			HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
							:TEST PRIORITY
731	005312	032737	000340	177776	5\$:	BIT	#340,CC
732	005320	001405			BEQ	6\$	
	005322	012737	000107	000302	MOV	#107,\$FATAL	:MOVE TO MAILBOX # ***** 107 *****
	005330	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	005332	000000			HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
733	005334	012706	000500		6\$:	MOV	#BUFF,SP
734	005340	012737	005356	000020	MOV	#7\$,RTRAP2	
735	005346	012737	000357	000022	MOV	#357,RTRAP2+2	:SET NEW 'CC' AND PRIORITY


```

736 005354 000004          IOT          ;TRAP HERE
737 005356          7$:      BMI          8$
      005356 100405          MOV          #110,$FATAL
      005360 012737 000110 000302  INC          (R2)
      005366 005212          HALT          ;MOVE TO MAILBOX # ***** 110 *****
      005370 000000          ;SET MSGTYP TO FATAL ERROR
      ;N NOT SET
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 707

738 005372          8$:      BEQ          9$
      005372 001405          MOV          #111,$FATAL
      005374 012737 000111 000302  INC          (R2)
      005402 005212          HALT          ;MOVE TO MAILBOX # ***** 111 *****
      005404 000000          ;SET MSGTYP TO FATAL ERROR
      ;Z NOT SET
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 701

739 005406          9$:      BVS          10$
      005406 102405          MOV          #112,$FATAL
      005410 012737 000112 000302  INC          (R2)
      005416 005212          HALT          ;MOVE TO MAILBOX # ***** 112 *****
      005420 000000          ;SET MSGTYP TO FATAL ERROR
      ;V NOT SET
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 673

740 005422          10$:     BCS          11$
      005422 103405          MOV          #113,$FATAL
      005424 012737 000113 000302  INC          (R2)
      005432 005212          HALT          ;MOVE TO MAILBOX # ***** 113 *****
      005434 000000          ;SET MSGTYP TO FATAL ERROR
      ;C NOT SET
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 665

741 005436 013706 177776 11$:     MOV          CC,SP
742 005442 042706 000017    BIC          #17,SP
743 005446 022706 000340    CMP          #340,SP
744 005452 001405          BEQ          13$
      005454          12$:     MOV          #114,$FATAL
      005454 012737 000114 000302  INC          (R2)
      005462 005212          HALT          ;MOVE TO MAILBOX # ***** 114 *****
      005464 000000          ;SET MSGTYP TO FATAL ERROR
      ;PRIORITY WAS CHANGED,OR WRONG $STNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 651

745 005466 012737 000022 000020 13$:     MOV          #22,20
746 005474 005037 000022    CLR          22
      ;.+2
      ;HALT
    
```


747

```
.SBTTL TEST #25 - TEST THAT A TRAP OCCURS ON AN EMT INS
:*****
:TEST 25 - TEST THAT A TRAP OCCURS ON AN EMT INS
:*****
TST25: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;CPU POWER BIT FOUND SET
      BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #25,$TESTN ;SEQUENCE ERROR?
      BNE TST26-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
748 005544 012706 000500 MOV #BUFF,SP ;STACK POINTER SETUP
749 005550 012737 005572 000030 MOV #1$,RTRAP3 ;RETURN LOCATION
750 005556 104000 EMT ;RESERVE INSTRUCTION, SHOULD TRAP
751 005560 012737 000115 000302 MOV #115,$FATAL ;MOVE TO MAILBOX # ***** 115 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;EMT DIDN'T TRAP,OR WRONG $STNM
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 764
```

752 005572

1\$:

753

.SBTTL TEST #26 - TEST DEC OF STACK POINTER ON A TRAP OPER

:TEST 26 - TEST DEC OF STACK POINTER ON A TRAP OPER

005572	032737	000001	177766	TST26:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005600	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
005602	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
005610	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005612	000000				HALT		:CPU POWER BIT FOUND SET
005614	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
005622	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
005626	022737	000026	000304		CMP	#26,\$TESTN	:SEQUENCE ERROR?
005634	001011				BNE	TST27-12	:BRANCH TO ERROR HALT ON SEQ ERROR
754 005636	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
755 005642	012737	005652	000030		MOV	#1\$,RTRAP3	:RETURN POINTER
756 005650	104000				EMT		:RESERVED INSTRUCTION
757 005652	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
758 005656	001405				BEQ	TST27	
005660	012737	000116	000302		MOV	#116,\$FATAL	:MOVE TO MAILBOX # ***** 116 *****
005666	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005670	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

759

.SBTTL TEST #27 - TEST THAT PROPER P.C. IS SAVED

 ;TEST 27 - TEST THAT PROPER P.C. IS SAVED

005672	032737	000001	177766	TST27:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
005700	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
005702	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
005710	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005712	000000				HALT		:CPU POWER BIT FOUND SET
005714	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
005722	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
005726	022737	000027	000304		CMP	#27,\$TESTN	:SEQUENCE ERROR?
005734	001012				BNE	TST30-12	:BRANCH TO ERROR HALT ON SEQ ERROR
760 005736	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
761 005742	012737	005752	000030		MOV	#1\$,RTRAP3	:RTURN FROM TRAP POINTER
762 005750	104000				EMT		:TRAP ON THIS INSTRUCTION
763 005752	022737	005752	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK FOR INCREMENTED P.C.
764 005760	001405				BEQ	TST30	
005762	012737	000117	000302		MOV	#117,\$FATAL	:MOVE TO MAILBOX # ***** 117 *****
005770	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
005772	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

765

.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
 :*****

```

005774 032737 000001 177766 TST30: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006002 001410 BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
006004 012737 000177 000302 MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
006012 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
006014 000000 HALT ;CPU POWER BIT FOUND SET
006016 042737 000001 177766 BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
006024 005237 000304 100$: INC $TESTN ;UPDATE TEST NUMBER
006030 022737 000030 000304 CMP #30,$TESTN ;SEQUENCE ERROR?
006036 001040 BNE TST31-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
766 006040 012706 000500 MOV #BUFF,SP ;SET UP
767 006044 012737 006062 000030 MOV #1$,RTRAP3 ;SET UP
768 006052 005037 177776 CLR CC ;CLEAR CC AND PRIORITY
769 006056 000257 CCC
770 006060 104000 EMT ;TRAP
771 006062 023727 000476 000000 1$: CMP BUFF-2,#0 ;TEST THAT OLD STATUS WENT TO STACK
772 006070 001405 BEQ 2$
006072 012737 000120 000302 MOV #120,$FATAL ;MOVE TO MAILBOX # ***** 120 *****
006100 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
006102 000000 HALT ;INCORRECT STATUS
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 755
773 006104 012706 000500 2$: MOV #BUFF,SP ;SET UP
774 006110 012737 006130 000030 MOV #3$,RTRAP3 ;SET UP
775 006116 012737 000357 177776 MOV #357,CC ;SET PRIORITY
776 006124 000277 SCC ;SET CC
777 006126 104000 EMT ;TRAP
778 006130 023727 000476 000357 3$: CMP BUFF-2,#357 ;COMPARES STATUS ON STACK
779 006136 001405 BEQ TST31
006140 012737 000121 000302 MOV #121,$FATAL ;MOVE TO MAILBOX # ***** 121 *****
006146 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
006150 000000 HALT ;INCORRECT STATUS ON STACK,OR WRONG $TSTNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 732

```


780

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

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

006152	032737	000001	177766	TST31:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006160	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
006162	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
006170	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006172	000000				HALT		:CPU POWER BIT FOUND SET
006174	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
006202	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
006206	022737	000031	000304		CMP	#31,\$TESTN	:SEQUENCE ERROR?
006214	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
006216	000137	006460			JMP	TST32-12	:JUMP TO ERROR HALT ON SEQ ERROR
006222				1000\$:			
781 006222	012706	000500			MOV	#BUFF,SP	
782 006226	012737	006242	000030		MOV	#1\$,RTRAP3	
783 006234	005037	000032			CLR	RTRAP3+2	:CLEAR FUTURE PRIORITY AND CC
784 006240	104000				EMT		
785 006242				1\$:			:TEST FOR 'C' CLEARED
786 006242	100005				BPL	2\$	
006244	012737	000122	000302		MOV	#122,\$FATAL	:MOVE TO MAILBOX # ***** 122 *****
006252	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006254	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
787 006256				2\$:			
006256	001005				BNE	3\$	
006260	012737	000123	000302		MOV	#123,\$FATAL	:MOVE TO MAILBOX # ***** 123 *****
006266	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006270	000000				HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
788 006272				3\$:			
006272	102005				BVC	4\$	
006274	012737	000124	000302		MOV	#124,\$FATAL	:MOVE TO MAILBOX # ***** 124 *****
006302	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006304	000000				HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
789 006306				4\$:			
006306	103005				BCC	5\$	
006310	012737	000125	000302		MOV	#125,\$FATAL	:MOVE TO MAILBOX # ***** 125 *****
006316	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006320	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
790 006322	032737	000340	177776	5\$:	BIT	#340,CC	:TEST PRIORITY
791 006330	001405				BEQ	6\$	
006332	012737	000126	000302		MOV	#126,\$FATAL	:MOVE TO MAILBOX # ***** 126 *****
006340	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006342	000000				HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
792 006344	012706	000500		6\$:	MOV	#BUFF,SP	
793 006350	012737	006366	000030		MOV	#7\$,RTRAP3	
794 006356	012737	000357	000032		MOV	#357,RTRAP3+2	:SET NEW 'CC' AND PRIORITY


```

795 006364 104000          EMT          ;TRAP HERE
796 006366          7$:      BMI          8$
    006366 100405          MOV          #127,$FATAL ;MOVE TO MAILBOX # ***** 127 *****
    006370 012737 000127 000302  INC          (R2)      ;SET MSGTYP TO FATAL ERROR
    006376 005212          HALT          ;N NOT SET
    006400 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 707

797 006402          8$:      BEQ          9$
    006402 001405          MOV          #130,$FATAL ;MOVE TO MAILBOX # ***** 130 *****
    006404 012737 000130 000302  INC          (R2)      ;SET MSGTYP TO FATAL ERROR
    006412 005212          HALT          ;Z NOT SET
    006414 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 701

798 006416          9$:      BVS          10$
    006416 102405          MOV          #131,$FATAL ;MOVE TO MAILBOX # ***** 131 *****
    006420 012737 000131 000302  INC          (R2)      ;SET MSGTYP TO FATAL ERROR
    006426 005212          HALT          ;V NOT SET
    006430 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 673

799 006432          10$:     BCS          11$
    006432 103405          MOV          #132,$FATAL ;MOVE TO MAILBOX # ***** 132 *****
    006434 012737 000132 000302  INC          (R2)      ;SET MSGTYP TO FATAL ERROR
    006442 005212          HALT          ;C NOT SET
    006444 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 665

800 006446 000257          11$:     CCC
801 006450 022737 000340 177776  CMP          #340,CC
802 006456 001405          BEQ          TST32
    006460 012737 000133 000302  MOV          #133,$FATAL ;MOVE TO MAILBOX # ***** 133 *****
    006466 005212          INC          (R2)      ;SET MSGTYP TO FATAL ERROR
    006470 000000          HALT          ;PRIORITY WAS CHANGED,OR WRONG $STNM
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 653
  
```


803

.SBTTL TEST #32 - TEST THAT ALL COMB EMT WILL CAUSE A TRAP

 :TEST 32 - TEST THAT ALL COMB EMT WILL CAUSE A TRAP

006472	032737	000001	177766	TST32:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006500	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
006502	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
006510	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006512	000000				HALT		:CPU POWER BIT FOUND SET
006514	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
006522	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
006526	022737	000032	000304		CMP	#32,\$TESTN	:SEQUENCE ERROR?
006534	001011				BNE	3\$:BRANCH TO ERROR HALT ON SEQ ERROR
804 006536	012737	104000	006556		MOV	#EMT,2\$:INITIALIZE BASE EMT INSTRUCTION
805 006544	012737	006572	000030		MOV	#4\$,30	:RETURN FROM TRAP TO 3\$
806 006552	012706	000500		1\$:	MOV	#BUFF,SP	:SET UP STACK POINTER
807 006556	104000			2\$:	EMT		:TRAP INST. WILL BE MODIFIED TO EMT+377
808 006560				3\$:			
006560	012737	000134	000302		MOV	#134,\$FATAL	:MOVE TO MAILBOX # ***** 134 *****
006566	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006570	000000				HALT		:PREVIOUS INST FAILED TO TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
809 006572	005237	006556		4\$:	INC	2\$:INCREMENT TRAP INSTRUCTION
810 006576	022737	104377	006556		CMP	#EMT+377,2\$:EMT+377 TO EMT?
811 006604	103362				BHIS	1\$:HAVE WE TESTED ALL
812							:YES
813 006606	012737	000032	000030		MOV	#32,30	:/.+
814 006614	005037	000032			CLR	32	:HALT

815

.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
 :*****

006620	032737	000001	177766	TST33:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006626	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
006630	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
006636	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006640	000000				HALT		:CPU POWER BIT FOUND SET
006642	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
006650	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
006654	022737	000033	000304		CMP	#33,\$TESTN	:SEQUENCE ERROR?
006662	001006				BNE	TST34-12	:BRANCH TO ERROR HALT ON SEQ ERROR
816 006664	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
817 006670	012737	006712	000014		MOV	#1\$,RTRAP4	:RETURN LOCATION
818 006676	000003				TRT		:RESERVED INSTRUCTION, SHOULD TRAP
819 006700	012737	000135	000302		MOV	#135,\$FATAL	:MOVE TO MAILBOX # ***** 135 *****
006706	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006710	000000				HALT		:TRT DIDN'T TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 764

820 006712

1\$:

821

.SBTTL TEST #34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

 :TEST 34 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

006712	032737	000001	177766	TST34:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
006720	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
006722	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
006730	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
006732	000000				HALT		:CPU POWER BIT FOUND SET
006734	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
006742	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
006746	022737	000034	000304		CMP	#34,\$TESTN	:SEQUENCE ERROR?
006754	001011				BNE	TST35-12	:BRANCH TO ERROR HALT ON SEQ ERROR
822 006756	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
823 006762	012737	006772	000014		MOV	#1\$,RTRAP4	:RETURN POINTER
824 006770	000003				TRT		:RESERVED INSTRUCTION
825 006772	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
826 006776	001405				BEQ	TST35	
007000	012737	000136	000302		MOV	#136,\$FATAL	:MOVE TO MAILBOX # ***** 136 *****
007006	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007010	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

827

.SBTTL TEST #35 - TEST THAT PROPER P.C. IS SAVED

:TEST 35 - TEST THAT PROPER P.C. IS SAVED

007012	032737	000001	177766	TST35:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007020	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
007022	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
007030	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007032	000000				HALT		:CPU POWER BIT FOUND SET
007034	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
007042	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
007046	022737	000035	000304		CMP	#35,\$TESTN	:SEQUENCE ERROR?
007054	001012				BNE	TST36-12	:BRANCH TO ERROR HALT ON SEQ ERROR
828 007056	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
829 007062	012737	007072	000014		MOV	#1\$,RTRAP4	:RETURN FROM TRAP POINTER
830 007070	000003				TRT		:TRAP ON THIS INSTRUCTION
831 007072	022737	007072	000474	1\$:	CMP	#,BUFF-4	:CHECK FOR INCREMENTED P.C.
832 007100	001405				BEQ	TST36	
007102	012737	000137	000302		MOV	#137,\$FATAL	:MOVE TO MAILBOX # ***** 137 *****
007110	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007112	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

833

.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

007114	032737	000001	177766	TST36:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007122	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
007124	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
007132	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007134	000000				HALT		:CPU POWER BIT FOUND SET
007136	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
007144	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
007150	022737	000036	000304		CMP	#36,\$TESTN	:SEQUENCE ERROR?
007156	001040				BNE	TST37-12	:BRANCH TO ERROR HALT ON SEQ ERROR
834	007160	012706	000500		MOV	#BUFF,SP	:SET UP
835	007164	012737	007202	000014	MOV	#1\$,RTRAP4	:SET UP
836	007172	005037	177776		CLR	CC	:CLEAR CC AND PRIORITY
837	007176	000257			CCC		
838	007200	000003			TRT		:TRAP
839	007202	023727	000476	000000	1\$:	CMP	BUFF-2,#0
840							:TEST THAT OLD STATUS WENT TO STACK
841	007210	001405				BEQ	2\$
	007212	012737	000140	000302		MOV	#140,\$FATAL
	007220	005212				INC	(R2)
	007222	000000				HALT	
							:MOVE TO MAILBOX # ***** 140 *****
							:SET MSGTYP TO FATAL ERROR
							:INCORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
842	007224	012706	000500		2\$:	MOV	#BUFF,SP
843	007230	012737	007250	000014		MOV	#3\$,RTRAP4
844	007236	012737	000357	177776		MOV	#357,CC
845	007244	000277				SCC	
846	007246	000003				TRT	
847	007250	023727	000476	000357	3\$:	CMP	BUFF-2,#357
848	007256	001405				BEQ	TST37
	007260	012737	000141	000302		MOV	#141,\$FATAL
	007266	005212				INC	(R2)
	007270	000000				HALT	
							:MOVE TO MAILBOX # ***** 141 *****
							:SET MSGTYP TO FATAL ERROR
							:INCORRECT STATUS ON STACK,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 732

849

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

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

007272	032737	000001	177766	TST37:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007300	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
007302	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
007310	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007312	000000				HALT		:CPU POWER BIT FOUND SET
007314	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
007322	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
007326	022737	000037	000304		CMP	#37,\$TESTN	:SEQUENCE ERROR?
007334	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
007336	000137	007604			JMP	15\$:JUMP TO ERROR HALT ON SEQ ERROR
007342				1000\$:			
850 007342	012706	000500			MOV	#BUFF,SP	
851 007346	012737	007362	000014		MOV	#4\$,RTRAP4	
852 007354	005037	000016			CLR	RTRAP4+2	:CLEAR FUTURE PRIORITY AND CC
853 007360	000003				TRT		
854 007362				4\$:			:TEST FOR 'C' CLEARED
855 007362	100005				BPL	5\$	
007364	012737	000142	000302		MOV	#142,\$FATAL	:MOVE TO MAILBOX # ***** 142 *****
007372	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007374	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
856 007376				5\$:			
007376	001005				BNE	6\$	
007400	012737	000143	000302		MOV	#143,\$FATAL	:MOVE TO MAILBOX # ***** 143 *****
007406	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007410	000000				HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
857 007412				6\$:			
007412	102005				BVC	7\$	
007414	012737	000144	000302		MOV	#144,\$FATAL	:MOVE TO MAILBOX # ***** 144 *****
007422	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007424	000000				HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
858 007426				7\$:			
007426	103005				BCC	8\$	
007430	012737	000145	000302		MOV	#145,\$FATAL	:MOVE TO MAILBOX # ***** 145 *****
007436	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007440	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
859 007442	032737	000340	177776	8\$:	BIT	#340,CC	:TEST PRIORITY
860 007450	001405				BEQ	9\$	
007452	012737	000146	000302		MOV	#146,\$FATAL	:MOVE TO MAILBOX # ***** 146 *****
007460	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007462	000000				HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
861 007464	012706	000500		9\$:	MOV	#BUFF,SP	
862 007470	012737	007506	000014		MOV	#10\$,RTRAP4	
863 007476	012737	000357	000016		MOV	#357,RTRAP4+2	:SET NEW 'CC' AND PRIORITY


```

864 007504 000003          TRT          ;TRAP HERE
865 007506          10$:          BMI          11$
      007506 100405          MOV          #147,$FATAL ;MOVE TO MAILBOX # ***** 147 *****
      007510 012737 000147 000302      INC          (R2)      ;SET MSGTYP TO FATAL ERROR
      007516 005212          HALT          ;N NOT SET
      007520 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 707

866 007522          11$:          BEQ          12$
      007522 001405          MOV          #150,$FATAL ;MOVE TO MAILBOX # ***** 150 *****
      007524 012737 000150 000302      INC          (R2)      ;SET MSGTYP TO FATAL ERROR
      007532 005212          HALT          ;Z NOT SET
      007534 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 701

867 007536          12$:          BVS          13$
      007536 102405          MOV          #151,$FATAL ;MOVE TO MAILBOX # ***** 151 *****
      007540 012737 000151 000302      INC          (R2)      ;SET MSGTYP TO FATAL ERROR
      007546 005212          HALT          ;V NOT SET
      007550 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 673

868 007552          13$:          BCS          14$
      007552 103405          MOV          #152,$FATAL ;MOVE TO MAILBOX # ***** 152 *****
      007554 012737 000152 000302      INC          (R2)      ;SET MSGTYP TO FATAL ERROR
      007562 005212          HALT          ;C NOT SET
      007564 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 665

869 007566 013706 177776 14$:          MOV          CC,SP
870 007572 042706 000017          BIC          #17,SP
871 007576 022706 000340          CMP          #340,SP
872 007602 001405          BEQ          16$
      007604          15$:          MOV          #153,$FATAL ;MOVE TO MAILBOX # ***** 153 *****
      007604 012737 000153 000302      INC          (R2)      ;SET MSGTYP TO FATAL ERROR
      007612 005212          HALT          ;PRIORITY WAS CHANGED,OR WRONG $STNM
      007614 000000          ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 651

873 007616 012737 000016 000014 16$:          MOV          #16,14
874 007624 005037 000016          CLR          16
875
876          ;PDP-11 ILLEGAL AND ADDRESS INSTRUCTION TEST
877          ;ALL INSTRUCTIONS THAT ARE ILLEGAL
878          ;SHOULD TRAP TO LOCATION 10, AND THE
879          ;PC THAT POINTS TO THE TRAPPING INSTRUCTION
880          ;SHOULD BE PLACED ON THE STACK
881

```


882

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

007630	032737	000001	177766	TST40:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007636	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
007640	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
007646	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
007650	000000				HALT		:CPU POWER BIT FOUND SET
007652	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
007660	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
007664	022737	000040	000304		CMP	#40,\$TESTN	:SEQUENCE ERROR?
007672	001006				BNE	TST41-12	:BRANCH TO ERROR HALT ON SEQ ERROR
883 007674	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
884 007700	012737	007722	000010		MOV	#1\$,RTRAP	:RETURN LOCATION
885 007706	000100				JMP	R0	:ILLEGAL INSTRUCTION, SHOULD TRAP
886 007710	012737	000154	000302		MOV	#154,\$FATAL	:MOVE TO MAILBOX # ***** 154 *****
	007716	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	007720	000000			HALT		:ILLEGAL INSTRUCTION DIDN'T TRAP,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 764

887 007722

1\$:

888

.SBTTL TEST #41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

:TEST 41 - TEST DEC OF STACK POINTER ON A TRAP OPERATION

007722	032737	000001	177766	TST41:	BIT	#1,CPUERR	;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
007730	001410				BEQ	100\$;BRANCH AROUND CLEAR AND HALT IF CLEAR
007732	012737	000177	000302		MOV	#177,\$FATAL	;MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
007740	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
007742	000000				HALT		;CPU POWER BIT FOUND SET
007744	042737	000001	177766		BIC	#1,CPUERR	;CLEAR THE BIT FOUND SET
007752	005237	000304		100\$:	INC	\$TESTN	;UPDATE TEST NUMBER
007756	022737	000041	000304		CMP	#41,\$TESTN	;SEQUENCE ERROR?
007764	001011				BNE	TST42-12	;BRANCH TO ERROR HALT ON SEQ ERROR
889 007766	012706	000500			MOV	#BUFF,SP	;STACK POINTER SETUP
890 007772	012737	010002	000010		MOV	#1\$,RTRAP	;RETURN POINTER
891 010000	000100				JMP	R0	;RESERVED INSTRUCTION
892 010002	020627	000474		1\$:	CMP	SP,#BUFF-4	;TEST DECREMENT OF SP
893 010006	001405				BEQ	TST42	
010010	012737	000155	000302		MOV	#155,\$FATAL	;MOVE TO MAILBOX # ***** 155 *****
010016	005212				INC	(R2)	;SET MSGTYP TO FATAL ERROR
010020	000000				HALT		;NOT DECREMENTED TWO WORDS,OR WRONG \$STSTM
							;TO SCOPE REPLACE HALT WITH 240
							;AND REPLACE NEXT INST WITH 761

894

.SBTTL TEST #42 - TEST THAT PROPER P.C. IS SAVED

:TEST 42 - TEST THAT PROPER P.C. IS SAVED

010022	032737	000001	177766	TST42:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010030	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
010032	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
010040	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010042	000000				HALT		:CPU POWER BIT FOUND SET
010044	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
010052	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
010056	022737	000042	000304		CMP	#42,\$TESTN	:SEQUENCE ERROR?
010064	001012				BNE	TST43-12	:BRANCH TO ERROR HALT ON SEQ ERROR
895 010066	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
896 010072	012737	010102	000010		MOV	#1\$,RTRAP	:RETURN FROM TRAP POINTER
897 010100	000100				JMP	R0	:TRAP ON THIS INSTRUCTION
898 010102	022737	010102	000474	1\$:	CMP	#,BUFF-4	:CHECK FOR INCREMENTED P.C.
899 010110	001405				BEQ	TST43	
010112	012737	000156	000302		MOV	#156,\$FATAL	:MOVE TO MAILBOX # ***** 156 *****
010120	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010122	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

900

.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
 :*****

010124	032737	000001	177766	TST43:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010132	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
010134	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
010142	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010144	000000				HALT		:CPU POWER BIT FOUND SET
010146	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
010154	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
010160	022737	000043	000304		CMP	#43,\$TESTN	:SEQUENCE ERROR?
010166	001040				BNE	TST44-12	:BRANCH TO ERROR HALT ON SEQ ERROR
901 010170	012706	000500			MOV	#BUFF,SP	:SET UP
902 010174	012737	010212	000010		MOV	#1\$,RTRAP	:SET UP
903 010202	005037	177776			CLR	CC	:CLEAR CC AND PRIORITY
904 010206	000257				CCC		
905 010210	000100				JMP	R0	:TRAP
906 010212	023727	000476	000000	1\$:	CMP	BUFF-2,#0	:TEST THAT OLD STATUS WENT TO STACK
907 010220	001405				BEQ	2\$	
010222	012737	000157	000302		MOV	#157,\$FATAL	:MOVE TO MAILBOX # ***** 157 *****
010230	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010232	000000				HALT		:INCORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
908 010234	012706	000500		2\$:	MOV	#BUFF,SP	:SET UP
909 010240	012737	010260	000010		MOV	#3\$,RTRAP	:SET UP
910 010246	012737	000357	177776		MOV	#357,CC	:SET PRIORITY
911 010254	000277				SCC		:SET CC
912 010256	000100				JMP	R0	:TRAP
913 010260	023727	000476	000357	3\$:	CMP	BUFF-2,#357	:COMPARES STATUS ON STACK
914 010266	001405				BEQ	TST44	
010270	012737	000160	000302		MOV	#160,\$FATAL	:MOVE TO MAILBOX # ***** 160 *****
010276	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010300	000000				HALT		:INCORRECT STATUS ON STACK,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 732

915

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

010302	032737	000001	177766	TST44:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010310	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
010312	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
010320	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010322	000000				HALT		:CPU POWER BIT FOUND SET
010324	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
010332	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
010336	022737	000044	000304		CMP	#44,\$TESTN	:SEQUENCE ERROR?
010344	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
010346	000137	010610			JMP	TST45-12	:JUMP TO ERROR HALT ON SEQ ERROR
010352				1000\$:			
916	010352	012706	000500		MOV	#BUFF,SP	
917	010356	012737	010372	000010	MOV	#1\$,RTRAP	
918	010364	005037	000012		CLR	RTRAP+2	:CLEAR FUTURE PRIORITY AND CC
919	010370	000100			JMP	R0	
920	010372			1\$:			:TEST FOR 'C' CLEARED
921	010372	100005			BPL	2\$	
	010374	012737	000161	000302	MOV	#161,\$FATAL	:MOVE TO MAILBOX # ***** 161 *****
	010402	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	010404	000000			HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
922	010406			2\$:			
	010406	001005			BNE	3\$	
	010410	012737	000162	000302	MOV	#162,\$FATAL	:MOVE TO MAILBOX # ***** 162 *****
	010416	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	010420	000000			HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
923	010422			3\$:			
	010422	102005			BVC	4\$	
	010424	012737	000163	000302	MOV	#163,\$FATAL	:MOVE TO MAILBOX # ***** 163 *****
	010432	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	010434	000000			HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
924	010436			4\$:			
	010436	103005			BCC	5\$	
	010440	012737	000164	000302	MOV	#164,\$FATAL	:MOVE TO MAILBOX # ***** 164 *****
	010446	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	010450	000000			HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
925	010452	032737	000357	177776	5\$:	BIT	#357,CC
926	010460	001405			BEQ	6\$	
	010462	012737	000165	000302	MOV	#165,\$FATAL	:MOVE TO MAILBOX # ***** 165 *****
	010470	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	010472	000000			HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
927	010474	012706	000500		6\$:	MOV	#BUFF,SP
928	010500	012737	010516	000010	MOV	#7\$,RTRAP	
929	010506	012737	000357	000012	MOV	#357,RTRAP+2	:SET NEW 'CC' AND PRIORITY


```

930 010514 000100          JMP      R0          ;TRAP HERE
931 010516          7$:  BMI      8$
    010516 100405          MOV      #166,$FATAL ;MOVE TO MAILBOX # ***** 166 *****
    010520 012737 000166 000302  INC      (R2)        ;SET MSGTYP TO FATAL ERROR
    010526 005212          HALT                    ;N NOT SET
    010530 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 707

932 010532          8$:  BEQ      9$
    010532 001405          MOV      #167,$FATAL ;MOVE TO MAILBOX # ***** 167 *****
    010534 012737 000167 000302  INC      (R2)        ;SET MSGTYP TO FATAL ERROR
    010542 005212          HALT                    ;Z NOT SET
    010544 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 701

933 010546          9$:  BVS      10$
    010546 102405          MOV      #170,$FATAL ;MOVE TO MAILBOX # ***** 170 *****
    010550 012737 000170 000302  INC      (R2)        ;SET MSGTYP TO FATAL ERROR
    010556 005212          HALT                    ;V NOT SET
    010560 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 673

934 010562         10$:  BCS      11$
    010562 103405          MOV      #171,$FATAL ;MOVE TO MAILBOX # ***** 171 *****
    010564 012737 000171 000302  INC      (R2)        ;SET MSGTYP TO FATAL ERROR
    010572 005212          HALT                    ;C NOT SET
    010574 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 665

935 010576 013706 177776 11$:  MOV      CC,$P
936 010602 022706 000357  CMP      #357,$P
937 010606 001405          BEQ      TST45
    010610 012737 000172 000302  MOV      #172,$FATAL ;MOVE TO MAILBOX # ***** 172 *****
    010616 005212          INC      (R2)        ;SET MSGTYP TO FATAL ERROR
    010620 000000          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $TSTNM
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 653
  
```


938

.SBTTL TEST #45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
 :*****
 :TEST 45 - TEST THAT A TRAP OCCURES ON ALL ILLEGAL INS
 :*****

010622 032737 000001 177766
 010630 001410
 010632 012737 000177 000302
 010640 005212
 010642 000000
 010644 042737 000001 177766
 010652 005237 000304
 010656 022737 000045 000304
 010664 001006
 939 010666 012706 000500
 940 010672 012737 010714 000010
 941 010700 004000
 942 010702 012737 000173 000302
 010710 005212
 010712 000000

TST45: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
 BEQ 100\$;BRANCH AROUND CLEAR AND HALT IF CLEAR
 MOV #177,\$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
 INC (R2) ;SET MSGTYP TO FATAL ERROR
 HALT ;CPU POWER BIT FOUND SET
 BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
 100\$: INC \$TESTN ;UPDATE TEST NUMBER
 CMP #45,\$TESTN ;SEQUENCE ERROR?
 BNE TST46-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
 MOV #BUFF,SP ;STACK POINTER SETUP
 MOV #1\$,RTRAP ;RETURN LOCATION
 JSR R0,R0 ;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

943 010714

1\$:

944

.SBTTL TEST #46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
 :*****
 :TEST 46 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
 :*****

010714	032737	000001	177766	TST46:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
010722	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
010724	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
010732	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
010734	000000				HALT		:CPU POWER BIT FOUND SET
010736	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
010744	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
010750	022737	000046	000304		CMP	#46,\$TESTN	:SEQUENCE ERROR?
010756	001011				BNE	TST47-12	:BRANCH TO ERROR HALT ON SEQ ERROR
945 010760	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
946 010764	012737	010774	000010		MOV	#1\$,RTRAP	:RETURN POINTER
947 010772	004000				JSR	R0,R0	:RESERVED INS
948 010774	020627	000474		1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
949 011000	001405				BEQ	TST47	
011002	012737	000174	000302		MOV	#174,\$FATAL	:MOVE TO MAILBOX # ***** 174 *****
011010	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011012	000000				HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

950

.SBTTL TEST #47 - TEST THAT PROPER P.C. IS SAVED

 :TEST 47 - TEST THAT PROPER P.C. IS SAVED

011014	032737	000001	177766	TST47:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011022	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
011024	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
011032	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011034	000000				HALT		:CPU POWER BIT FOUND SET
011036	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
011044	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
011050	022737	000047	000304		CMP	#47,\$TESTN	:SEQUENCE ERROR?
011056	001012				BNE	TST50-12	:BRANCH TO ERROR HALT ON SEQ ERROR
951 011060	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
952 011064	012737	011074	000010		MOV	#1\$,RTRAP	:RETURN FROM TRAP POINTER
953 011072	004000				JSR	RO,RO	:TRAP ON THIS INS
954 011074	022737	011074	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK FOR INCREMENTED P.C.
955 011102	001405				BEQ	TST50	
011104	012737	000175	000302		MOV	#175,\$FATAL	:MOVE TO MAILBOX # ***** 175 *****
011112	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011114	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

956

957

.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

011116	032737	000001	177766	TST50:	BIT	#1, CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011124	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
011126	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
011134	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011136	000000				HALT		:CPU POWER BIT FOUND SET
011140	042737	000001	177766		BIC	#1, CPUERR	:CLEAR THE BIT FOUND SET
011146	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
011152	022737	000050	000304		CMP	#50,\$TESTN	:SEQUENCE ERROR?
011160	001040				BNE	TST51-12	:BRANCH TO ERROR HALT ON SEQ ERROR
958 011162	012706	000500			MOV	#BUFF,SP	:SET UP
959 011166	012737	011204	000010		MOV	#1\$,RTRAP	:SET UP
960 011174	005037	177776			CLR	CC	:CLEAR CC AND PRIORITY
961 011200	000257				CCC		
962 011202	004000				JSR	R0,R0	:TRAP
963 011204	023727	000476	000000	1\$:	CMP	BUFF-2,#0	:TEST THAT OLD STATUS WENT TO STACK
964 011212	001405				BEQ	2\$	
011214	012737	000176	000302		MOV	#176,\$FATAL	:MOVE TO MAILBOX # ***** 176 *****
011222	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011224	000000				HALT		:INCORRECT STATUS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 755
965 011226	012706	000500		2\$:	MOV	#BUFF,SP	:SET UP
966 011232	012737	011252	000010		MOV	#3\$,RTRAP	:SET UP
967 011240	012737	000357	177776		MOV	#357,CC	:SET PRIORITY
968 011246	000277				SCC		:SET CC
969 011250	004000				JSR	R0,R0	:TRAP
970 011252	023727	000476	000357	3\$:	CMP	BUFF-2,#357	:COMPARES STATUS ON STACK
971 011260	001405				BEQ	TST51	
011262	012737	000177	000302		MOV	#177,\$FATAL	:MOVE TO MAILBOX # ***** 177 *****
011270	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011272	000000				HALT		:INCORRECT STATUS ON STACK,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 732

972

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

 :TEST 51 - TEST THAT 'NEW' STATUS IS CORRECT

011274	032737	000001	177766	TST51:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011302	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
011304	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
011312	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011314	000000				HALT		:CPU POWER BIT FOUND SET
011316	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
011324	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
011330	022737	000051	000304		CMP	#51,\$TESTN	:SEQUENCE ERROR?
011336	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
011340	000137	011600			JMP	TST52-12	:JUMP TO ERROR HALT ON SEQ ERROR
011344				1000\$:			
973	011344	012706	000500		MOV	#BUFF,SP	
974	011350	012737	011364	000010	MOV	#1\$,RTRAP	
975	011356	005037	000012		CLR	RTRAP+2	:CLEAR FUTURE PRIORITY AND CC
976	011362	004000			JSR	R0,R0	
977	011364			1\$:			:TEST FOR 'C' CLEARED
978	011364	100005			BPL	2\$	
	011366	012737	000200	000302	MOV	#200,\$FATAL	:MOVE TO MAILBOX # ***** 200 *****
	011374	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	011376	000000			HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
979	011400			2\$:			
	011400	001005			BNE	3\$	
	011402	012737	000201	000302	MOV	#201,\$FATAL	:MOVE TO MAILBOX # ***** 201 *****
	011410	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	011412	000000			HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 753
980	011414			3\$:			
	011414	102005			BVC	4\$	
	011416	012737	000202	000302	MOV	#202,\$FATAL	:MOVE TO MAILBOX # ***** 202 *****
	011424	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	011426	000000			HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 745
981	011430			4\$:			
	011430	103005			BCC	5\$	
	011432	012737	000203	000302	MOV	#203,\$FATAL	:MOVE TO MAILBOX # ***** 203 *****
	011440	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	011442	000000			HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
							:TEMP STORAGE
982	011444	013700	177776	5\$:	MOV	CC,R0	
983	011450	001405			BEQ	6\$	
	011452	012737	000204	000302	MOV	#204,\$FATAL	:MOVE TO MAILBOX # ***** 204 *****
	011460	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	011462	000000			HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 727
984	011464	012706	000500	6\$:	MOV	#BUFF,SP	
985	011470	012737	011506	000010	MOV	#7\$,RTRAP	
986	011476	012737	000357	000012	MOV	#357,RTRAP+2	:SET NEW 'CC' AND PRIORITY


```

987 011504 004000          JSR      R0,R0          ;TRAP HERE
988 011506          7$:      BMI      8$
    011506 100405          MOV      #205,$FATAL      ;MOVE TO MAILBOX # ***** 205 *****
    011510 012737 000205 000302  INC      (R2)          ;SET MSGTYP TO FATAL ERROR
    011516 005212          HALT                    ;N NOT SET
    011520 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 710

989 011522          8$:      BEQ      9$
    011522 001405          MOV      #206,$FATAL      ;MOVE TO MAILBOX # ***** 206 *****
    011524 012737 000206 000302  INC      (R2)          ;SET MSGTYP TO FATAL ERROR
    011532 005212          HALT                    ;Z NOT SET
    011534 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 702

990 011536          9$:      BVS      10$
    011536 102405          MOV      #207,$FATAL      ;MOVE TO MAILBOX # ***** 207 *****
    011540 012737 000207 000302  INC      (R2)          ;SET MSGTYP TO FATAL ERROR
    011546 005212          HALT                    ;V NOT SET
    011550 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 674

991 011552          10$:     BCS      11$
    011552 103405          MOV      #210,$FATAL      ;MOVE TO MAILBOX # ***** 210 *****
    011554 012737 000210 000302  INC      (R2)          ;SET MSGTYP TO FATAL ERROR
    011562 005212          HALT                    ;C NOT SET
    011564 000000          ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 666

992 011566 013700 177776 11$:  MOV      CC,R0
993 011572 022700 000357  CMP      #357,R0
994 011576 001405          BEQ      TST52
    011600 012737 000211 000302  MOV      #211,$FATAL      ;MOVE TO MAILBOX # ***** 211 *****
    011606 005212          INC      (R2)          ;SET MSGTYP TO FATAL ERROR
    011610 000000          HALT                    ;PRIORITY WAS CHANGED,OR WRONG $STNM
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 654

995
  
```


996

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

011612	032737	000001	177766	TST52:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
011620	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
011622	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
011630	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011632	000000				HALT		:CPU POWER BIT FOUND SET
011634	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
011642	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
011646	022737	000052	000304		CMP	#52,\$TESTN	:SEQUENCE ERROR?
011654	001040				BNE	3\$:BRANCH TO ERROR HALT ON SEQ ERROR
997 011656	005037	177766			CLR	CPUERR	:CLEAR CPU ERROR REGISTER
998 011662	005737	177766			TST	CPUERR	:VERIFY THAT IT CLEARED
999 011666	001405				BEQ	1\$	
011670	012737	000212	000302		MOV	#212,\$FATAL	:MOVE TO MAILBOX # ***** 212 *****
011676	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011700	000000				HALT		:CPU ERROR REG FAILED TO CLEAR
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 765
1000 011702	012706	000500		1\$:	MOV	#BUFF,SP	:STACK POINTER SETUP
1001 011706	012737	011732	000004		MOV	#2\$,RTRAP5	:RETURN LOCATION
1002 011714	005737	000001			TST	1	:ILLEGAL ADDRESS INS, SHOULD TRAP
1003 011720	012737	000213	000302		MOV	#213,\$FATAL	:MOVE TO MAILBOX # ***** 213 *****
011726	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011730	000000				HALT		:ILLEGAL ADDRESS DID NOT TRAP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 751
1004 011732				2\$:			
1005 011732	013737	177766	000502		MOV	CPUERR,RCPUER	:READ AND SAVE CPU ERROR REGISTER
1006 011740	042737	177413	000502		BIC	#CERMSK,RCPUER	:MASK OFF UNUSED BITS OF CPU ERROR REG
1007 011746	022737	000100	000502		CMP	#100,RCPUER	:ODD ADDRESS BIT SET?
1008 011754	001405				BEQ	4\$	
011756				3\$:			
011756	012737	000214	000302		MOV	#214,\$FATAL	:MOVE TO MAILBOX # ***** 214 *****
011764	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
011766	000000				HALT		:INCORRECT CPU ERROR REG CONTENTS, OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 732
1009 011770	005037	177766		4\$:	CLR	CPUERR	:CLEAR ODD ADDRESS BIT
1010							

1011

.SBTTL TEST #53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
 :*****
 :TEST 53 - TEST DEC OF STACK POINTER ON A TRAP OPERATION
 :*****

011774	032737	000001	177766	TST53:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012002	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
012004	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
012012	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012014	000000				HALT		:CPU POWER BIT FOUND SET
012016	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
012024	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
012030	022737	000053	000304		CMP	#53,\$TESTN	:SEQUENCE ERROR?
012036	001012				BNE	TST54-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1012	012040	012706	000500		MOV	#BUFF,SP	:STACK POINTER SETUP
1013	012044	012737	012056	000004	MOV	#1\$,RTRAP5	:RETURN POINTER
1014	012052	005737	000001		TST	1	:RESERVED INS
1015	012056	020627	000474	1\$:	CMP	SP,#BUFF-4	:TEST DECREMENT OF SP
1016	012062	001405			BEQ	TST54	
	012064	012737	000215	000302	MOV	#215,\$FATAL	:MOVE TO MAILBOX # ***** 215 *****
	012072	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	012074	000000			HALT		:NOT DECREMENTED TWO WORDS,OR WRONG \$TSTNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760

1017

.SBTTL TEST #54 - TEST THAT PROPER P.C. IS SAVED

:TEST 54 - TEST THAT PROPER P.C. IS SAVED

012076	032737	000001	177766	TST54:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012104	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
012106	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
012114	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012116	000000				HALT		:CPU POWER BIT FOUND SET
012120	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
012126	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
012132	022737	000054	000304		CMP	#54,\$TESTN	:SEQUENCE ERROR?
012140	001013				BNE	TST55-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1018 012142	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
1019 012146	012737	012160	000004		MOV	#1\$,RTRAP5	:RETURN FROM TRAP POINTER
1020 012154	005737	000001			TST	1	:TRAP ON THIS INSTRUCTION
1021 012160	022737	012160	000474	1\$:	CMP	#1\$,BUFF-4	:CHECK FOR INCREMENTED P.C.
1022 012166	001405				BEQ	TST55	
012170	012737	000216	000302		MOV	#216,\$FATAL	:MOVE TO MAILBOX # ***** 216 *****
012176	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012200	000000				HALT		:INCORRECT P.C.,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 757

1023

```

.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: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
      BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
      MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;CPU POWER BIT FOUND SET
      BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #55,$TESTN ;SEQUENCE ERROR?
      BNE TST56-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
1024: MOV #BUFF,SP ;SET UP
1025: MOV #1$,RTRAP5 ;SET UP
1026: CLR CC ;CLEAR CC AND PRIORITY
1027: CCC
1028: TST 1 ;TRAP
1029: CMP BUFF-2,#0 ;TEST THAT OLD STATUS WENT TO STACK
1030: BEQ 2$
      MOV #217,$FATAL ;MOVE TO MAILBOX # ***** 217 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;INCORRECT STATUS
      ;TO SCOPE REPLACE HALT WITH 240
      ;AND REPLACE NEXT INST WITH 754
2$: MOV #BUFF,SP ;SET UP
      MOV #3$,RTRAP5 ;SET UP
      MOV #357,CC ;SET PRIORITY
      SCC ;SET CC
      TST 1 ;TRAP
3$: CMP BUFF-2,#357 ;COMPARES STATUS ON STACK
      BEQ TST56
      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
    
```

012202 032737 000001 177766
 012210 001410
 012212 012737 000177 000302
 012220 005212
 012222 000000
 012224 042737 000001 177766
 012232 005237 000304
 012236 022737 000055 000304
 012244 001042
 1024 012246 012706 000500
 1025 012252 012737 012272 000004
 1026 012260 005037 177776
 1027 012264 000257
 1028 012266 005737 000001
 1029 012272 023727 000476 000000
 1030 012300 001405
 012302 012737 000217 000302
 012310 005212
 012312 000000

 1031 012314 012706 000500
 1032 012320 012737 012342 000004
 1033 012326 012737 000357 177776
 1034 012334 000277
 1035 012336 005737 000001
 1036 012342 023727 000476 000357
 1037 012350 001405
 012352 012737 000220 000302
 012360 005212
 012362 000000

1038

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

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

012364	032737	000001	177766	TST56:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012372	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
012374	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
012402	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012404	000000				HALT		:CPU POWER BIT FOUND SET
012406	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
012414	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
012420	022737	000056	000304		CMP	#56,\$TESTN	:SEQUENCE ERROR?
012426	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
012430	000137	012676			JMP	TST57-12	:JUMP TO ERROR HALT ON SEQ ERROR
012434				1000\$:			
1039 012434	012706	000500			MOV	#BUFF,SP	
1040 012440	012737	012456	000004		MOV	#1\$,RTRAP5	
1041 012446	005037	000006			CLR	RTRAP5+2	:CLEAR FUTURE PRIORITY AND CC
1042 012452	005737	000001			TST	1	:TRAP HERE
1043 012456				1\$:			:TEST FOR 'C' CLEARED
1044 012456	100005				BPL	2\$	
012460	012737	000221	000302		MOV	#221,\$FATAL	:MOVE TO MAILBOX # ***** 221 *****
012466	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012470	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 760
1045 012472				2\$:			
012472	001005				BNE	3\$	
012474	012737	000222	000302		MOV	#222,\$FATAL	:MOVE TO MAILBOX # ***** 222 *****
012502	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012504	000000				HALT		:Z NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 752
1046 012506				3\$:			
012506	102005				BVC	4\$	
012510	012737	000223	000302		MOV	#223,\$FATAL	:MOVE TO MAILBOX # ***** 223 *****
012516	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012520	000000				HALT		:V NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 744
1047 012522				4\$:			
012522	103005				BCC	5\$	
012524	012737	000224	000302		MOV	#224,\$FATAL	:MOVE TO MAILBOX # ***** 224 *****
012532	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012534	000000				HALT		:C NOT CLEARED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 736
1048 012536	032737	000357	177776	5\$:	BIT	#357,CC	:TEST PRIORITY FOR ZERO
1049 012544	001405				BEQ	6\$	
012546	012737	000225	000302		MOV	#225,\$FATAL	:MOVE TO MAILBOX # ***** 225 *****
012554	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012556	000000				HALT		:PRIORITY NOT ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 725
1050 012560	012706	000500		6\$:	MOV	#BUFF,SP	
1051 012564	012737	012604	000004		MOV	#7\$,RTRAP5	
1052 012572	012737	000357	000006		MOV	#357,RTRAP5+2	:SET NEW 'CC' AND PRIORITY

1061

.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

012710	032737	000001	177766	TST57:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
012716	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
012720	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
012726	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
012730	000000				HALT		:CPU POWER BIT FOUND SET
012732	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
012740	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
012744	022737	000057	000304		CMP	#57,\$TESTN	:SEQUENCE ERROR?
012752	001027				BNE	2\$:BRANCH TO ERROR HALT ON SEQ ERROR
1062	012754	005037	177766		CLR	CPUERR	:CLEAR CPU ERROR REGISTER
1063	012760	012706	000150		MOV	#150,R6	:R6 = 150
1064	012764	012737	013006	000004	MOV	#1\$,4	:STACK OVERFLOW TRAP POINTER
1065	012772	005746			TST	-(R6)	:WITH R6 = 150 SHOULD TRAP
1066	012774	012737	000233	000302	MOV	#233,\$FATAL	:MOVE TO MAILBOX # ***** 233 *****
	013002	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	013004	000000			HALT		:SHOULD HAVE TRAPPED,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 762
1067	013006			1\$:			
1068	013006	013737	177766	000502	MOV	CPUERR,RCPUER	:SAVE CPU ERROR REGISTER
1069	013014	042737	177413	000502	BIC	#CERMSK,RCPUER	:MASK OFF UNUSED CPU ERROR REG BITS
1070	013022	022737	000004	000502	CMP	#4,RCPUER	:IS YELLOW ZONE BIT SET?
1071	013030	001405			BEQ	3\$	
	013032						
	013032	012737	000234	000302	2\$:	MOV	#234,\$FATAL
	013040	005212			INC	(R2)	:MOVE TO MAILBOX # ***** 234 *****
	013042	000000			HALT		:SET MSGTYP TO FATAL ERROR
							:INCORRECT CPU ERROR REGISTER CONTENTS, OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 743
1072	013044	005037	177766	3\$:	CLR	CPUERR	:CLEAR YELLOW ZONE BIT

1073

.SBTTL TEST #60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP

 :TEST 60 - TEST FOR DEC OF R6 ON OVERFLOW TRAP

013050	032737	000001	177766	TST60:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013056	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
013060	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
013066	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
013070	000000				HALT		:CPU POWER BIT FOUND SET
013072	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
013100	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
013104	022737	000060	000304		CMP	#60,\$TESTN	:SEQUENCE ERROR?
013112	001011				BNE	TST61-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1074	013114	012706	000150		MOV	#150,R6	:R6 = 150
1075	013120	012737	013130	000004	MOV	#1\$,4	:TRAP POINTER
1076	013126	005746			TST	-(R6)	:WITH R6 = 150 SHOULD TRAP
1077	013130	020627	000142	1\$:	CMP	R6,#142	:DID R6 DECREMENT
1078	013134	001405			BEQ	TST61	
	013136	012737	000235	000302	MOV	#235,\$FATAL	:MOVE TO MAILBOX # ***** 235 *****
	013144	005212			INC		:SET MSGTYP TO FATAL ERROR
	013146	000000			HALT		:R6 NOT = 142,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761

1106

.SBTTL TEST #62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP

 :TEST 62 - TEST THAT AN 10 CAUSES AN OVERFLOW TRAP

013334	032737	000001	177766	TST62:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013342	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
013344	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
013352	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
013354	000000				HALT		:CPU POWER BIT FOUND SET
013356	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
013364	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
013370	022737	000062	000304		CMP	#62,\$TESTN	:SEQUENCE ERROR?
013376	001011				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
013400	012706	000400			MOV	#400,R6	:SET UP STACK TO OVERFLOW
013404	012737	013422	000010		MOV	#1\$,10 ;SET UP	10 VECTOR
013412	012737	013434	000004		MOV	#2\$,4	:SET UP OVERFLOW VECTOR
013420	000010				10		:THIS TRAP SHOULD CAUSE OVERFLOW
013422				1\$:			
013422	012737	000241	000302		MOV	#241,\$FATAL	:MOVE TO MAILBOX # ***** 241 *****
013430	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
013432	000000				HALT		:TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
013434	012737	000012	000010	2\$:	MOV	#10+2,10	

1107

.SBTTL TEST #63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP

 :TEST 63 - TEST THAT AN IOT CAUSES AN OVERFLOW TRAP

013442	032737	000001	177766	TST63:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013450	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
013452	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
013460	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
013462	000000				HALT		:CPU POWER BIT FOUND SET
013464	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
013472	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
013476	022737	000063	000304		CMP	#63,\$TESTN	:SEQUENCE ERROR?
013504	001011				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
013506	012706	000400			MOV	#400,R6	:SET UP STACK TO OVERFLOW
013512	012737	013530	000020		MOV	#1\$,20 ;SET UP	IOT VECTOR
013520	012737	013542	000004		MOV	#2\$,4	:SET UP OVERFLOW VECTOR
013526	000004				IOT		:THIS TRAP SHOULD CAUSE OVERFLOW
013530				1\$:			
013530	012737	000242	000302		MOV	#242,\$FATAL	:MOVE TO MAILBOX # ***** 242 *****
013536	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
013540	000000				HALT		:TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
013542	012737	000022	000020	2\$:	MOV	#20+2,20	

1108

```

.SBTTL TEST #64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
:*****
:TEST 64 - TEST THAT AN EMT CAUSES AN OVERFLOW TRAP
:*****
013550 032737 000001 177766 TST64: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013556 001410 BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
013560 012737 000177 000302 MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013566 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
013570 000000 HALT ;CPU POWER BIT FOUND SET
013572 042737 000001 177766 BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
013600 005237 000304 100$: INC $TESTN ;UPDATE TEST NUMBER
013604 022737 000064 000304 CMP #64,$TESTN ;SEQUENCE ERROR?
013612 001011 BNE 1$ ;BRANCH TO ERROR HALT ON SEQ ERROR
013614 012706 000400 MOV #400,R6 ;SET UP STACK TO OVERFLOW
013620 012737 013636 000030 MOV #1$,30 ;SET UP EMT VECTOR
013626 012737 013650 000004 MOV #2$,4 ;SET UP OVERFLOW VECTOR
013634 104000 EMT ;THIS TRAP SHOULD CAUSE OVERFLOW
013636 012737 000243 000302 1$: MOV #243,$FATAL ;MOVE TO MAILBOX # ***** 243 *****
013644 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
013646 000000 HALT ;TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 761

013650 012737 000032 000030 2$: MOV #30+2,30
  
```


1109

.SBTTL TEST #65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
 :*****
 :TEST 65 - TEST THAT AN TRAP CAUSES AN OVERFLOW TRAP
 :*****

```

013656 032737 000001 177766 TST65: BIT #1,CPUERR :SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013664 001410 BEQ 100$ :BRANCH AROUND CLEAR AND HALT IF CLEAR
013666 012737 000177 000302 MOV #177,$FATAL :MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
013674 005212 INC (R2) :SET MSGTYP TO FATAL ERROR
013676 000000 HALT :CPU POWER BIT FOUND SET
013700 042737 000001 177766 BIC #1,CPUERR :CLEAR THE BIT FOUND SET
013706 005237 000304 100$: INC $TESTN :UPDATE TEST NUMBER
013712 022737 000065 000304 CMP #65,$TESTN :SEQUENCE ERROR?
013720 001011 BNE 1$ :BRANCH TO ERROR HALT ON SEQ ERROR
013722 012706 000400 MOV #400,R6 :SET UP STACK TO OVERFLOW
013726 012737 013744 000034 MOV #1$,34 ;SET UP TRAP VECTOR
013734 012737 013756 000004 MOV #2$,4 :SET UP OVERFLOW VECTOR
013742 104400 TRAP :THIS TRAP SHOULD CAUSE OVERFLOW
013744 1$:
013744 012737 000244 000302 MOV #244,$FATAL :MOVE TO MAILBOX # ***** 244 *****
013752 005212 INC (R2) :SET MSGTYP TO FATAL ERROR
013754 000000 HALT :TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG $STNM
:TO SCOPE REPLACE HALT WITH 240
:AND REPLACE NEXT INST WITH 761

013756 012737 000036 000034 2$: MOV #34+2,34
    
```


1110

.SBTTL TEST #66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP

:TEST 66 - TEST THAT AN TRT CAUSES AN OVERFLOW TRAP

013764	032737	000001	177766	TST66:	BIT	#1,CPUERR	:	SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
013772	001410				BEQ	100\$:	BRANCH AROUND CLEAR AND HALT IF CLEAR
013774	012737	000177	000302		MOV	#177,\$FATAL	:	MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014002	005212				INC	(R2)	:	SET MSGTYP TO FATAL ERROR
014004	000000				HALT		:	CPU POWER BIT FOUND SET
014006	042737	000001	177766		BIC	#1,CPUERR	:	CLEAR THE BIT FOUND SET
014014	005237	000304		100\$:	INC	\$TESTN	:	UPDATE TEST NUMBER
014020	022737	000066	000304		CMP	#66,\$TESTN	:	SEQUENCE ERROR?
014026	001011				BNE	1\$:	BRANCH TO ERROR HALT ON SEQ ERROR
014030	012706	000400			MOV	#400,R6	:	SET UP STACK TO OVERFLOW
014034	012737	014052	000014		MOV	#1\$,14 ;SET UP	:	TRT VECTOR
014042	012737	014064	000004		MOV	#2\$,4	:	SET UP OVERFLOW VECTOR
014050	000003				TRT		:	THIS TRAP SHOULD CAUSE OVERFLOW
014052				1\$:				
014052	012737	000245	000302		MOV	#245,\$FATAL	:	MOVE TO MAILBOX # ***** 245 *****
014060	005212				INC	(R2)	:	SET MSGTYP TO FATAL ERROR
014062	000000				HALT		:	TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG \$STNM
							:	TO SCOPE REPLACE HALT WITH 240
							:	AND REPLACE NEXT INST WITH 761
014064	012737	000016	000014	2\$:	MOV	#14+2,14		

1111

.SBTTL TEST #67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP

:TEST 67 - TEST THAT AN ILLA CAUSES AN OVERFLOW TRAP

014072	032737	000001	177766	TST67:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014100	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014102	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014110	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014112	000000				HALT		:CPU POWER BIT FOUND SET
014114	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014122	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014126	022737	000067	000304		CMP	#67,\$TESTN	:SEQUENCE ERROR?
014134	001011				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
014136	012706	000400			MOV	#400,R6	:SET UP STACK TO OVERFLOW
014142	012737	014160	000010		MOV	#1\$,10 ;SET UP	ILLA VECTOR
014150	012737	014172	000004		MOV	#2\$,4	:SET UP OVERFLOW VECTOR
014156	004700				ILLA		:THIS TRAP SHOULD CAUSE OVERFLOW
014160				1\$:			
014160	012737	000246	000302		MOV	#246,\$FATAL	:MOVE TO MAILBOX # ***** 246 *****
014166	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014170	000000				HALT		:TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
				2\$:			
1112 014172	012737	000012	000010		MOV	#10+2,10	:STACK PUSHED FOUR WORDS?
1113 014200	020627	000370			CMP	R6,#370	
1113 014204	001405				BEQ	T\$170	
014206	012737	000247	000302		MOV	#247,\$FATAL	:MOVE TO MAILBOX # ***** 247 *****
014214	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014216	000000				HALT		:CORRECT # (4) OF WORDS WERE NOT PUSHED ONTO STACK
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 746

1114

.SBTTL TEST #70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
 :*****
 :TEST 70 - TEST THAT AN ILLB CAUSES AN OVERFLOW TRAP
 :*****

014220	032737	000001	177766	TST70:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014226	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014230	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014236	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014240	000000				HALT		:CPU POWER BIT FOUND SET
014242	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014250	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014254	022737	000070	000304		CMP	#70,\$TESTN	:SEQUENCE ERROR?
014262	001011				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
014264	012706	000400			MOV	#400,R6	:SET UP STACK TO OVERFLOW
014270	012737	014306	000010		MOV	#1\$,10 ;SET UP	ILLB VECTOR
014276	012737	014320	000004		MOV	#2\$,4	:SET UP OVERFLOW VECTOR
014304	000100				ILLB		:THIS TRAP SHOULD CAUSE OVERFLOW
014306				1\$:			
014306	012737	000250	000302		MOV	#250,\$FATAL	:MOVE TO MAILBOX # ***** 250 *****
014314	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014316	000000				HALT		:TRAP FLAG OVERFLOW DID NOT OCCUR,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 761
1115	014320	012737	000012	000010	2\$:	MOV	#10+2,10

1116

.SBTTL TEST #71 - TEST FOR FALSE OVERFLOW TRAP

 :TEST 71 - TEST FOR FALSE OVERFLOW TRAP

014326	032737	000001	177766	TST71:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014334	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014336	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014344	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014346	000000				HALT		:CPU POWER BIT FOUND SET
014350	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014356	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014362	022737	000071	000304		CMP	#71,\$TESTN	:SEQUENCE ERROR?
014370	001023				BNE	1\$:BRANCH TO ERROR HALT ON SEQ ERROR
1117							
1118	014372	012737	014440	000004	MOV	#1\$,4	:SET UP OVERFLOW POINTER
1119	014400	012706	001002		MOV	#1002,R6	
1120	014404	005746			TST	-(R6)	:SHOULD NOT OVERFLOW
1121	014406	012706	002002		MOV	#2002,R6	
1122	014412	005746			TST	-(R6)	:SHOULD NOT OVERFLOW
1123	014414	012706	004002		MOV	#4002,R6	
1124	014420	005746			TST	-(R6)	:SHOULD NOT OVERFLOW
1125	014422	012706	010002		MOV	#10002,R6	
1126	014426	005746			TST	-(R6)	:SHOULD NOT OVERFLOW
1127	014430	012706	020000		MOV	#20000,R6	
1128	014434	005746			TST	-(R6)	
1129	014436	000405			BR	2\$	
	014440			1\$:			
	014440	012737	000251	000302	MOV	#251,\$FATAL	:MOVE TO MAILBOX # ***** 251 *****
	014446	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	014450	000000			HALT		:STACK OVERFLOWED,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 747
1130	014452	012737	000006	000004	2\$:	MOV	#6,4
1131	014460	005037	000006		CLR	6	

1132

.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

014464	032737	000001	177766	TST72:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014472	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014474	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014502	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014504	000000				HALT		:CPU POWER BIT FOUND SET
014506	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014514	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014520	022737	000072	000304		CMP	#72,\$TESTN	:SEQUENCE ERROR?
014526	001013				BNE	TST73-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1133	014530	012706	000500		MOV	#BUFF,SP	
1134	014534	012737	014570	000014	MOV	#2\$,RTRAP4	:SET UP TO TRAP TO 14
1135	014542	012746	000020		MOV	#20,-(SP)	:PUSH T BIT
1136	014546	012746	014554		MOV	#1\$,-(SP)	:PUSH PC
1137	014552	000002			RTI		:SET T BIT
1138	014554	000240		1\$:	NOP		:TRAP HERE
1139	014556	012737	000252	000302	MOV	#252,\$FATAL	:MOVE TO MAILBOX # ***** 252 *****
	014564	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	014566	000000			HALT		:TRACE BIT DID NOT TRAP!,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 757
1140	014570			2\$:			

1141

.SBTTL TEST #73 - TEST STACK POINTER DECREMENTS

 :TEST 73 - TEST STACK POINTER DECREMENTS

014570	032737	000001	177766	TST73:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014576	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014600	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014606	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014610	000000				HALT		:CPU POWER BIT FOUND SET
014612	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014620	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014624	022737	000073	000304		CMP	#73,\$TESTN	:SEQUENCE ERROR?
014632	001023				BNE	TST74-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1142	014634	012706	000500		MOV	#BUFF,SP	
1143	014640	012737	014674	000014	MOV	#2\$,RTRAP4	
1144	014646	012746	000020		MOV	#20,-(SP)	:PUSH T BIT
1145	014652	012746	014660		MOV	#1\$,-(SP)	:PUSH PC
1146	014656	000002			RTI		:SET T BIT
1147	014660	000240		1\$:	NOP		:TRAP HERE
1148	014662	012737	000253	000302	MOV	#253,\$FATAL	:MOVE TO MAILBOX # ***** 253 *****
	014670	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	014672	000000			HALT		:TRACE BIT DID NOT TRAP!
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 757
1149	014674	020627	000474	2\$:	CMP	SP,#BUFF-4	
1150	014700	001405			BEQ	TST74	
	014702	012737	000254	000302	MOV	#254,\$FATAL	:MOVE TO MAILBOX # ***** 254 *****
	014710	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	014712	000000			HALT		:STACK POINTER WAS NOT PUSHED BY TRAP,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 747

1151

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

 :TEST 74 - TEST FOR PROPER PC ON STACK

014714	032737	000001	177766	TST74:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
014722	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
014724	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
014732	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
014734	000000				HALT		:CPU POWER BIT FOUND SET
014736	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
014744	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
014750	022737	000074	000304		CMP	#74,\$TESTN	:SEQUENCE ERROR?
014756	001016				BNE	TST75-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1152	014760	012706	000500		MOV	#BUFF,SP	
1153	014764	012737	015026	000014	MOV	#2\$,RTRAP4	
1154	014772	012746	000020		MOV	#20,-(SP)	:PUSH T BIT
1155	014776	012746	015004		MOV	#1\$,-(SP)	:PUSH PC
1156	015002	000002			RTI		:SET T BIT
1157	015004	022737	015004	000474	1\$:	CMP	#1\$,BUFF-4
1158	015012	001405				BEQ	TST75
	015014	012737	000255	000302		MOV	#255,\$FATAL
	015022	005212				INC	(R2)
	015024	000000				HALT	:MOVE TO MAILBOX # ***** 255 *****
							:SET MSGTYP TO FATAL ERROR
							:CORRECT PC WAS NOT SAVED ON STACK,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 754

1159 015026

2\$:

1160

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

 :TEST 75 - TEST THAT RTT POPS T- BIT

```

015026 032737 000001 177766 TST75: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
015034 001410 BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
015036 012737 000177 000302 MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
015044 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
015046 000000 HALT ;CPU POWER BIT FOUND SET
015050 042737 000001 177766 BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
015056 005237 000304 100$: INC $TESTN ;UPDATE TEST NUMBER
015062 022737 000075 000304 CMP #75,$TESTN ;SEQUENCE ERROR?
015070 001015 BNE TST76-12 ;BRANCH TO ERROR HALT ON SEQ ERROR
1161 015072 012706 000500 MOV #BUFF,SP
1162 015076 005001 CLR R1 ;CLEAR R1
1163 015100 012746 000020 MOV #20,-(SP)
1164 015104 012746 015120 MOV #1$,-(SP)
1165 015110 012737 015136 000014 MOV #2$,14
1166 015116 000006 RTT
1167 015120 000240 1$: NOP
1168 015122 001405 BEQ TST76
015124 012737 000256 000302 MOV #256,$FATAL ;MOVE TO MAILBOX # ***** 256 *****
015132 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
015134 000000 HALT ;T-BIT DID NOT TRAP,OR WRONG $TESTN
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 755
1169 015136 2$:
    
```


1170

.SBTTL TEST #76 - TEST THAT RTT ALLOWS ONE INST. BEFORE TRAP

 :TEST 76 - TEST THAT RTT ALLOWS ONE INST. BEFORE TRAP

015136	032737	000001	177766	TST76:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
015144	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
015146	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
015154	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
015156	000000				HALT		:CPU POWER BIT FOUND SET
015160	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
015166	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
015172	022737	000076	000304		CMP	#76,\$TESTN	:SEQUENCE ERROR?
015200	001031				BNE	TST77-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1171	015202	012705	177777		MOV	#177777,R5	
1172	015206	012706	000500	1\$:	MOV	#BUFF,SP	
1173	015212	012746	000020		MOV	#20,-(SP)	
1174	015216	012746	015234		MOV	#2\$,-(SP)	
1175	015222	012737	015254	000014	MOV	#3\$,14	
1176	015230	005001			CLR	R1	:CLEAR R0
1177	015232	000006			RTT		:SET T-BIT
1178	015234	005201		2\$:	INC	R1	
1179	015236	005205			INC	R5	
1180	015240	001762			BEQ	1\$:DO THIS TEST NO MORE THAN 2 TIMES
1181	015242	012737	000257	000302	MOV	#257,\$FATAL	:MOVE TO MAILBOX # ***** 257 *****
	015250	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	015252	000000			HALT		:DID NOT TRAP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 752
							:SEE IF RTT ALLOWS 1 INST.
1182	015254	005301		3\$:	DEC	R1	
1183	015256	001407			BEQ	4\$	
1184	015260	005205			INC	R5	:DO THIS TEST NO MORE THAN TWO TIMES
1185	015262	001751			BEQ	1\$	
	015264	012737	000260	000302	MOV	#260,\$FATAL	:MOVE TO MAILBOX # ***** 260 *****
	015272	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	015274	000000			HALT		:RTT DID NOT ALLOW 1 INST.,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 741
1186	015276			4\$:			

1187

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

 :TEST 77 - TEST THAT RTI DOES NOT ALLOW 1 INST.

015276	032737	000001	177766	TST77:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
015304	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
015306	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
015314	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
015316	000000				HALT		:CPU POWER BIT FOUND SET
015320	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
015326	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
015332	022737	000077	000304		CMP	#77,\$TESTN	:SEQUENCE ERROR?
015340	001023				BNE	TST100-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1188	015342	012706	000500		MOV	#BUFF,SP	
1189	015346	012746	000020		MOV	#20,-(SP)	
1190	015352	012746	015370		MOV	#1\$,-(SP)	
1191	015356	012737	015404	000014	MOV	#2\$,14	
1192	015364	005001			CLR	R1	
1193	015366	000002			RTI		:SET T-BIT
1194	015370	005201		1\$:	INC	R1	:RTI SHOULD NOT ALLOW THIS
1195	015372	012737	000261	000302	MOV	#261,\$FATAL	:MOVE TO MAILBOX # ***** 261 *****
	015400	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	015402	000000			HALT		:T- BIT DID NOT CAUSE TRAP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 756
1196	015404	005701		2\$:	TST	R1	:RTI SHOULD NOT ALLOW 1 INST. BEFORE TRAP
1197							
1198	015406	001405			BEQ	TST100	
	015410	012737	000262	000302	MOV	#262,\$FATAL	:MOVE TO MAILBOX # ***** 262 *****
	015416	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	015420	000000			HALT		:RTI DID ALLOW 1 INST. BEFORE TRAP,OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 747

1199

.SBTTL TEST #100 - DOES THE PROCESSOR TRAP WHEN R7 IS ODD?

 :TEST 100 - DOES THE PROCESSOR TRAP WHEN R7 IS ODD?

015422	032737	000001	177766	TST100:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
015430	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
015432	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
015440	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
015442	000000				HALT		:CPU POWER BIT FOUND SET
015444	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
015452	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
015456	022737	000100	000304		CMP	#100,\$TESTN	:SEQUENCE ERROR?
015464	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
015466	000137	015732			JMP	TST101-12	:JUMP TO ERROR HALT ON SEQ ERROR
015472				1000\$:			
1200	015472	012706	000500		MOV	#BUFF,R6	:SET UP STACK POINTER
1201	015476	012737	015522	000004	MOV	#1\$,4	:RETURN FROM TRAP
1202	015504	012707	000001		MOV	#1,R7	:PC EQUALS ONE
1203	015510	012737	000263	000302	MOV	#263,\$FATAL	:MOVE TO MAILBOX # ***** 263 *****
	015516	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	015520	000000			HALT		:ODD ADDRESS SHOULD HAVE TRAPPED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 763
1204	015522	022737	000001	000474	1\$:	CMP	#1,BUFF-4
1205	015530	001405				BEQ	2\$
	015532	012737	000264	000302		MOV	#264,\$FATAL
	015540	005212				INC	(R2)
	015542	000000				HALT	
							:MOVE TO MAILBOX # ***** 264 *****
							:SET MSGTYP TO FATAL ERROR
							:CORRECT PC WAS NOT SAVED ON STACK
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 752
1206	015544	012706	000500		2\$:	MOV	#BUFF,R6
1207	015550	012737	015572	000004		MOV	#4\$,4
1208	015556	005207				INC	R7
1209	015560				3\$:		
	015560	012737	000265	000302		MOV	#265,\$FATAL
	015566	005212				INC	(R2)
	015570	000000				HALT	
							:MOVE TO MAILBOX # ***** 265 *****
							:SET MSGTYP TO FATAL ERROR
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 737
1210	015572	022737	015561	000474	4\$:	CMP	#3\$+1,BUFF-4
1211	015600	001405				BEQ	5\$
	015602	012737	000266	000302		MOV	#266,\$FATAL
	015610	005212				INC	(R2)
	015612	000000				HALT	
							:MOVE TO MAILBOX # ***** 266 *****
							:SET MSGTYP TO FATAL ERROR
							:CORRECT PC NOT ON STACK
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 726
1212	015614	012706	000500		5\$:	MOV	#BUFF,R6
1213	015620	012737	015642	000004		MOV	#7\$,4
1214	015626	005307			6\$:	DEC	R7
1215	015630	012737	000267	000302		MOV	#267,\$FATAL
	015636	005212				INC	(R2)
	015640	000000				HALT	
							:MAKE PC ODD
							:MOVE TO MAILBOX # ***** 267 *****
							:SET MSGTYP TO FATAL ERROR
							:SHOULD TRAP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 713
							:CHECK VALUE OF PC ON STACK
1216	015642	022737	015627	000474	7\$:	CMP	#6\$+1,BUFF-4
1217	015650	001405				BEQ	8\$
	015652	012737	000270	000302		MOV	#270,\$FATAL
							:MOVE TO MAILBOX # ***** 270 *****

1226

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

015744	032737	000001	177766	TST101:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
015752	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
015754	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
015762	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
015764	000000				HALT		:CPU POWER BIT FOUND SET
015766	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
015774	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
016000	022737	000101	000304		CMP	#101,\$TESTN	:SEQUENCE ERROR?
016006	001027				BNE	3\$:BRANCH TO ERROR HALT ON SEQ ERROR
1227	016010	012706	000500		MOV	#BUFF,R6	
1228	016014	012737	016052	000014	MOV	#1\$,14	:TRACE TRAP
1229	016022	005027	000016		CLR	#16	
1230	016026	005027	000022		CLR	#22	
1231	016032	012737	016100	000020	MOV	#4\$,20	:IOT TRAP
1232	016040	012746	000020		MOV	#20,-(SP)	:PUSH T BIT
1233	016044	012746	016052		MOV	#1\$,-(SP)	:PUSH PC
1234	016050	000006			RTT		
1235	016052	000004		1\$:	IOT		:TRAP, NEW CC HAVE TRACE RESET
1236	016054			2\$:			
	016054	012737	000273	000302	MOV	#273,\$FATAL	:MOVE TO MAILBOX # ***** 273 *****
	016062	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	016064	000000			HALT		:TRACE TRAP WAS NOT INHIBITED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 750
1237	016066			3\$:			
	016066	012737	000274	000302	MOV	#274,\$FATAL	:MOVE TO MAILBOX # ***** 274 *****
	016074	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	016076	000000			HALT		:WRONG TSTNM,OR WRONG \$TSTNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 743
1238	016100	012737	000016	000014	4\$:	MOV	#16,14
1239	016106	012737	000022	000020	MOV	#22,20	

1240

.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
 :*****

```

016114 032737 000001 177766 TST102: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
016122 001410 BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
016124 012737 000177 000302 MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
016132 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
016134 000000 HALT ;CPU POWER BIT FOUND SET
016136 042737 000001 177766 BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
016144 005237 000304 100$: INC $TESTN ;UPDATE TEST NUMBER
016150 022737 000102 000304 CMP #102,$TESTN ;SEQUENCE ERROR?
016156 001020 BNE 2$ ;BRANCH TO ERROR HALT ON SEQ ERROR
1241 016160 012706 000500 MOV #BUFF,R6 ;SET UP STACK POINTER
1242 016164 012737 016232 000014 MOV #3$,14 ;TRACE TRAP RETURN
1243 016172 005037 000016 CLR 16
1244 016176 012746 000020 MOV #20,-(SP) ;SET THE T BIT
1245 016202 012746 016210 MOV #1$,-(SP)
1246 016206 000002 RTI
1247 016210 033727 000476 000020 1$: BIT BUFF-2,#20 ;CHECK FOR T BIT ON STACK
1248 016216 001005 BNE 3$
016220 2$:
016220 012737 000275 000302 MOV #275,$FATAL ;MOVE TO MAILBOX # ***** 275 *****
016226 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
016230 000000 HALT ;T BIT NOT SAVED ON THE STACK, OR WRONG $STNM
;TO SCOPE REPLACE HALT WITH 240
;AND REPLACE NEXT INST WITH 752
1249 016232 012737 000016 000014 3$: MOV #16,14
1250
1251
1252
1253
```

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

1254

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

 :TEST 103 - TEST NON-EXISTENT ADDRESS TRAPS

016240	032737	000001	177766	TST103:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
016246	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
016250	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
016256	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
016260	000000				HALT		:CPU POWER BIT FOUND SET
016262	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
016270	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
016274	022737	000103	000304		CMP	#103,\$TESTN	:SEQUENCE ERROR?
016302	001402				BEQ	1000\$:BRANCH OVER ERROR HALT ON SEQ ERROR JUMP IF OK
016304	000137	016650			JMP	18\$:JUMP TO ERROR HALT ON SEQ ERROR
016310				1000\$:			

1255

;THIS ROUTINE TESTS MEMORY UNTIL IT DOES A NXM TRAP

1256					BR	5\$	
1257	016310	000402					
1258	016312	000000		3\$:		0	
1259	016314	000000		4\$:		0	
1260	016316	005000		5\$:	CLR	R0	
1261	016320	005037	177766		CLR	CPUERR	:CLEAR CPU ERROR REGISTER
1262	016324	005037	000006		CLR	6	
1263	016330	012737	016364	000004	MOV	#8\$,4	:SET UP ADDRESS TRAP ENTRANCE
1264	016336	012706	000500	6\$:	MOV	#BUFF,SP	
1265	016342	105720			TSTB	(R0)+	:IF OUTSIDE OF CORE, TRAP TO 4
1266	016344	020027	160000		CMP	R0,#160000	:IS POINTER IN SIDE CORE
1267	016350	101772			BLOS	6\$:TEST THE REST OF CORE
1268	016352			7\$:			
	016352	012737	000276	000302	MOV	#276,\$FATAL	:MOVE TO MAILBOX # ***** 276 *****
	016360	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	016362	000000			HALT		:SHOULD HAVE TRAPPED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 751

1269

:RETURN HERE ON AN ADDRESS TRAP

1270	016364	010037	016314	000502	8\$:	MOV	R0,4\$:MOVE THE FIRST NXM LOCAVION IN 5\$
1271	016370	013737	177766	000502		MOV	CPUERR,RCPUER	:SAVE CPU ERROR REGISTER
1272	016376	042737	177413	000502		BIC	#CERMSK,RCPUER	:MASK OFF UNUSED ERROR REG BITS
1273	016404	020027	160000			CMP	R0,#160000	:WHICH CPU ERROR REG BIT SHOULD BE
1274								: SET - NON EXISTANT MEMORY (BIT 5)
1275								: OR UNIBUS TIMEOUT (BIT 4)
1276	016410	103012			BHIS	9\$:BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET	
1277	016412	022737	000040	000502	CMP	#40,RCPUER	:IS NON-EXISTANT MEMORY BIT SET?	
1278	016420	001417			BEQ	10\$		
	016422	012737	000277	000302	MOV	#277,\$FATAL	:MOVE TO MAILBOX # ***** 277 *****	
	016430	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	016432	000000			HALT		:INCORRECT CPU ERROR REG CONTENTS	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 725	

1279

1280	016436	022737	000020	000502	9\$:	BR	10\$:IS UNIBUS TIMEOUT BIT SET?
1281	016444	001405			CMP	#20,RCPUER		
	016446	012737	000300	000302	BEQ	10\$		
	016454	005212			MOV	#300,\$FATAL	:MOVE TO MAILBOX # ***** 300 *****	
	016456	000000			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	


```

1282          ;THIS ROUTINE DOES NXM TRAPS UNTIL IT FINDS AN EXISTANT MEMORY LOCATION
1283 016460 012700 160001 10$: MOV #160001,R0 ;SET UP THE HIGHEST MEM LOCATION
1284 016464 005037 177766 11$: CLR CPUERR ;CLEAR CPU ERROR REGISTER
1285 016470 012737 016530 000004 MOV #14$,4 ;SET UP THE VECTOR
1286 016476 012706 000500 MOV #BUFF,SP
1287 016502 105740 TSTB -(R0) ;DOES IT EXIST?
1288 016504 005200 13$: INC R0 ;IF YES INCREMENT IT
1289 016506 020037 016314 CMP R0,4$ ;IS IT THE SAME LOCATION?
1290 016512 001463 BEQ 19$
    016514 012737 000301 000302 MOV #301,$FATAL ;MOVE TO MAILBOX # ***** 301 *****
    016522 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016524 000000 HALT ;CONTENTS OF R0 AND 5$ 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

1291
1292
1293
1294 016526 000455 BR 19$
1295
1296 016530 005737 177776 14$: TST STATUS
1297 016534 001405 BEQ 15$
    016536 012737 000302 000302 MOV #302,$FATAL ;MOVE TO MAILBOX # ***** 302 *****
    016544 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016546 000000 HALT ;NEW PSW SHOULD HAVE BEEN ZERO
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 657

1298
1299 016550 013737 177766 000502 15$: MOV CPUERR,RCPUER ;SAVE CPU ERROR REGISTER
1300 016556 042737 177413 000502 BIC #CERMSK,RCPUER ;MASK OFF UNUSED ERROR REG BITS
1301 016564 020027 160000 CMP R0,#160000 ;WHICH CPU ERROR REG BIT SHOULD BE
1302 ; SET - NON EXISTANT MEMORY (BIT 5)
1303 ; OR UNIBUS TIOEOUT (BIT 4)
1304 016570 103012 BHIS 16$ ;BRANCH IF UNIBUS TIMEOUT BIT SHOULD BE SET
1305 016572 022737 000040 000502 CMP #40,RCPUER ;IS NON-EXISTENT MEMORY BIT SET?
1306 016600 001417 BEQ 17$
    016602 012737 000303 000302 MOV #303,$FATAL ;MOVE TO MAILBOX # ***** 303 *****
    016610 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016612 000000 HALT ;INCORRECT CPU ERROR REG CONTENTS
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 635

1307 016614 000411 BR 17$
1308 016616 022737 000020 000502 16$: CMP #20,RCPUER ;IS UNIBUS TIMEOUT BIT SET?
1309 016624 001405 BEQ 17$
    016626 012737 000304 000302 MOV #304,$FATAL ;MOVE TO MAILBOX # ***** 304 *****
    016634 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016636 000000 HALT ;INCORRECT CPU ERROR REG CONTENTS
    ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 623

1310 016640 023727 000474 016504 17$: CMP BUFF-4,#13$
1311 016646 001706 BEQ 11$
    016650 18$: MOV #305,$FATAL ;MOVE TO MAILBOX # ***** 305 *****
    016650 012737 000305 000302 INC (R2) ;SET MSGTYP TO FATAL ERROR
    016656 005212 HALT ;OLD PC WAS NOT SAVED OR WRONG $TESTN
    016660 000000 ;TO SCOPE REPLACE HALT WITH 240
    ;AND REPLACE NEXT INST WITH 612

1312 016662 012737 000006 000004 19$: MOV #6,4
    
```



```
1313 016670 005037 000006                    CLR        6  
1314                                            ;THIS ROUTINE WILL FIGURE OUT IF YOU HAVE A DL11W  
1315                                             
1316 016674 005037 016720                    CLR        PROFTE                    ;INITIALLY CLEAR THE LOCATION  
1317 016700 012706 000500                    MOV        #BUFF,SP                    ;SET UP THE STACK POINTER  
1318 016704 012737 016722 000004            MOV        #DL11W,4                    ;SET UP THE TRAP VECTOR  
1319 016712 005737 177564                    TST        TPS                        ;TEST THE PUNCH STATUS REGISTER  
1320 016716 000403                            BR        DL11W1                        ;BRANCH IF IT EXISTS  
1321 016720 000000                            PROFTE: .WORD 0  
1322 016722 005237 016720                    DL11W: INC    PROFTE                    ;INCREMENT IF NO DL11W  
1323 016726 012737 000006 000004            DL11W1: MOV   #6,4
```


1324

```

.SBTTL TEST #104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
:*****
:TEST 104 - TEST THAT A TTY INRUP CAUSES AN OVERFLOW TRAP
:*****
TST104: BIT #1,CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
        BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
        MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;CPU POWER BIT FOUND SET
        BIC #1,CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #104,$TESTN ;SEQUENCE ERROR?
      BNE 1$ ;BRANCH TO ERROR HALT ON SEQ ERROR
      TST PROFTE
      BNE 3$
      RESET
      MOV #340,STATUS ;LOCK OUT INTERRUPT
      MOV #400,R6 ;SET UP STACK TO OVERFLOW
      MOV #2$,4 ;SET UP OVERFLOW TRAP
      MOV #1$,64 ;SET UP INTERRUPT VECTOR
      MOV #100,TTCSR ;SET INTERRUPT ENABLE
      CLR STATUS ;ALLOW INTERRUPT TO OCCUR
      MOV #306,$FATAL ;MOVE TO MAILBOX # ***** 306 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;NO INTERRUPT OCCURRED
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 746

1335 017062 017062 012737 000307 000302 1$: MOV #307,$FATAL ;MOVE TO MAILBOX # ***** 307 *****
      INC (R2) ;SET MSGTYP TO FATAL ERROR
      HALT ;OVERFLOW TRAP DID NOT OCCUR OR WRONG $STNM
          ;TO SCOPE REPLACE HALT WITH 240
          ;AND REPLACE NEXT INST WITH 741

1336 017074 005037 177564 2$: CLR TTCSR ;CLEAR INTERRUPT ENABLE
1337 017100 012737 000006 000004 MOV #6,4
1338 017106 005037 000006 CLR 6
1339 017112 3$:
    
```

016734 032737 000001 177766
 016742 001410
 016744 012737 000177 000302
 016752 005212
 016754 000000
 016756 042737 000001 177766
 016764 005237 000304
 016770 022737 000104 000304
 016776 001031
 1325 017000 005737 016720
 1326 017004 001042
 1327 017006 000005
 1328 017010 012737 000340 177776
 1329 017016 012706 000400
 1330 017022 012737 017074 000004
 1331 017030 012737 017062 000064
 1332 017036 012737 000100 177564
 1333 017044 005037 177776
 1334 017050 012737 000306 000302
 017056 005212
 017060 000000

 1335 017062
 017062 012737 000307 000302
 017070 005212
 017072 000000

 1336 017074 005037 177564
 1337 017100 012737 000006 000004
 1338 017106 005037 000006
 1339 017112

1340

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

017112	032737	000001	177766	TST105:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
017120	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
017122	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
017130	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
017132	000000				HALT		:CPU POWER BIT FOUND SET
017134	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
017142	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
017146	022737	000105	000304		CMP	#105,\$TESTN	:SEQUENCE ERROR?
017154	001037				BNE	2\$:BRANCH TO ERROR HALT ON SEQ ERROR
1341	017156	005737	016720		TST	PROFTE	
1342	017162	001046			BNE	4\$	
1343	017164	012706	000500		MOV	#BUFF,R6	
1344	017170	012737	000340	177776	MOV	#340,STATUS	:SET TO A HIGH PRIORITY LEVEL
1345	017176	012737	017242	000064	MOV	#1\$,64	
1346	017204	012737	000100	177564	MOV	#100,TTCSR	:INTERRUPT FOR TTY PUNCH/PRINTER
1347	017212	012737	017266	000034	MOV	#3\$,34	:TRAP VECTOR
1348	017220	012737	017254	000064	MOV	#2\$,64	:TTY VECTOR
1349	017226	012737	000340	000036	MOV	#340,36	:IF TRAP TRAPS, MOVE 340 TO PRIORITY
1350	017234	005037	177776		CLR	STATUS	:SHOULD INTERRUPT AT END OF CLR INST
1351	017240	104400			TRAP		:TTY INTERRUPT SHOULD OVERRIDE TRAP
1352	017242			1\$:			
	017242	012737	000310	000302	MOV	#310,\$FATAL	:MOVE TO MAILBOX # ***** 310 *****
	017250	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	017252	000000			HALT		:NEITHER TRAP NOR INRUPT OCCURED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 740
1353	017254			2\$:			
	017254	012737	000311	000302	MOV	#311,\$FATAL	:MOVE TO MAILBOX # ***** 311 *****
	017262	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	017264	000000			HALT		:INRUPT OCCURRED FIRST,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 733
1354	017266	005037	000036	3\$:	CLR	36	
1355	017272	042737	000100	177564	BIC	#100,TTCSR	
1356	017300			4\$:			

1357

.SBTTL TEST #106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
 :*****
 :TEST 106 - TEST THAT A PENDING INRUP, INRUP BETWEEN TRAPS
 :*****

017300	032737	000001	177766	TST106:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET	
017306	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR	
017310	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL	
017316	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR	
017320	000000				HALT		:CPU POWER BIT FOUND SET	
017322	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET	
017330	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER	
017334	022737	000106	000304		CMP	#106,\$TESTN	:SEQUENCE ERROR?	
017342	001031				BNE	2\$:BRANCH TO ERROR HALT ON SEQ ERROR	
1358	017344	005737	016720		TST	PROFTE		
1359	017350	001046			BNE	4\$		
1360	017352	012706	000500		MOV	#BUFF,R6		
1361	017356	012737	000340	177776	MOV	#340,STATUS		
1362	017364	012737	000100	177564	MOV	#100,TTCSR		
1363	017372	012737	017424	000034	MOV	#1\$,34	:TRAP	
1364	017400	012737	017440	000064	MOV	#3\$,64	:TTY OUTPUT	
1365	017406	012737	017426	000020	MOV	#2\$,20	:IOT	
1366	017414	012737	000340	000022	MOV	#340,22	:IOT PRIORITY	
1367	017422	104400			TRAP		:THE ACT OF TRAPPING LOWER PRIORITY	
1368	017424	000004			IOT		:INTERRUPT SHOULD OCCUR IN PLACE OF IOT TRAP	
1369	017426							
	017426	012737	000312	000302	2\$:	MOV	#312,\$FATAL	:MOVE TO MAILBOX # ***** 312 *****
	017434	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
	017436	000000				HALT		:NO INTERRUPT BETWEEN TRAPS,OR WRONG \$STNM
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 741
1370	017440	005037	000022		3\$:	CLR	22	:CLR IOT PRIORITY
1371	017444	012737	000036	000034		MOV	#36,34	
1372	017452	012737	000066	000064		MOV	#66,64	
1373	017460	012737	000022	000020		MOV	#22,20	
1374	017466				4\$:			
1375								

1376

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

:TEST 107 - TEST THAT 'RESET' GOES TO OUTSIDE WORLD

017466	032737	000001	177766	TST107:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
017474	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
017476	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
017504	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
017506	000000				HALT		:CPU POWER BIT FOUND SET
017510	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
017516	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
017522	022737	000107	000304		CMP	#107,\$TESTN	:SEQUENCE ERROR?
017530	001027				BNE	TST110-12	:BRANCH TO ERROR HALT ON SEQ ERROR
1377	017532	005737	016720		TST	PROFTE	
1378	017536	001031			BNE	2\$	
1379	017540	012737	000100	177564	MOV	#100,TTCSR	:SET INTERRUPT ENABLE
1380	017546	012737	000100	177560	MOV	#100,TRCSR	:SET INTERRUPT ENABLE
1381	017554	000005			RESET		:SHOULD CLEAR INTERRUPT ENABLE
1382	017556	032737	000100	177564	BIT	#100,TTCSR	:TEST FOR CLEAR
1383	017564	001405			BEQ	1\$	
	017566	012737	000313	000302	MOV	#313,\$FATAL	:MOVE TO MAILBOX # ***** 313 *****
	017574	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	017576	000000			HALT		:RESET FAILED TO CLEAR TTCSR
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 754
							:TEST FOR CLEAR
1384	017600	032737	000100	177560	1\$:	BIT	#100,TRCSR
1385	017606	001405				BEQ	TST110
	017610	012737	000314	000302		MOV	#314,\$FATAL
	017616	005212				INC	(R2)
	017620	000000				HALT	
							:MOVE TO MAILBOX # ***** 314 *****
							:SET MSGTYP TO FATAL ERROR
							:RESET FAILED TO CLEAR TRCSR,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 743
1386	017622				2\$:		

1387

.SBTTL TEST #110 - TEST THAT RESET HAS NO EFFECT ON TRACE TRAP
 :*****
 :TEST 110 - TEST THAT RESET HAS NO EFFECT ON TRACE TRAP
 :*****

017622	032737	000001	177766	TST110:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
017630	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
017632	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
017640	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
017642	000000				HALT		:CPU POWER BIT FOUND SET
017644	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
017652	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
017656	022737	000110	000304		CMP	#110,\$TESTN	:SEQUENCE ERROR?
017664	001014				BNE	2\$:BRANCH TO ERROR HALT ON SEQ ERROR
1388	017666	012706	000500		MOV	#BUFF,R6	:SET STACK
1389	017672	012737	017730	000014	MOV	#3\$,14	:SET UP TRACE VECTOR
1390	017700	012746	000020		MOV	#20,-(R6)	:SET THE T-BIT ON STACK
1391	017704	012746	017712		MOV	#1\$,-(R6)	:MOVE NEW PC ON STACK
1392	017710	000006			RTT		
1393	017712	000005		1\$:	RESET		:SHOULD HAVE NO EFFECT
1394	017714	000005			RESET		:NO EFFECT
1395	017716			2\$:			
	017716	012737	000315	000302	MOV	#315,\$FATAL	:MOVE TO MAILBOX # ***** 315 *****
	017724	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	017726	000000			HALT		:TRACE TRAP FAILED,OR WRONG \$STNM
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 756
1396	017730	005037	177776	3\$:	CLR	STATUS	:CLEAR TRACK
1397	017734	005037	000016		CLR	16	:TRACE STATUS
1398	017740	012737	000016	000014	MOV	#16,14	
1399							

1400

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

 :TEST 111 - TEST THAT WHEN TTY INRUPTS IT POPS NEW STATUS

017746	032737	000001	177766	TST111:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET	
017754	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR	
017756	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL	
017764	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR	
017766	000000				HALT		:CPU POWER BIT FOUND SET	
017770	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET	
017776	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER	
020002	022737	000111	000304		CMP	#111,\$TESTN	:SEQUENCE ERROR?	
020010	001051				BNE	4\$:BRANCH TO ERROR HALT ON SEQ ERROR	
1401	020012	005737	016720		TST	PROFTE		
1402	020016	001055			BNE	6\$		
1403	020020	000005			RESET			
1404	020022	012706	000500		MOV	#BUFF,R6	:SET UP STACK	
1405	020026	012737	020052	000064	MOV	#1\$,64	:INTERRUPT VECTOR	
1406	020034	005037	177776		CLR	STATUS	:DROP PROCESSOR PRIORITY	
1407	020040	012737	000357	000066	MOV	#357,66	:HIGH PRIORITY ON INTERRUPT	
1408	020046	005137	177564		COM	TTCSR	:SHOULD SET INTERRUPT ENABLE & INTERRUPT	
1409	020052	023727	177776	000357	1\$:	CMP	STATUS,#357	
1410	020060	001405			BEQ	2\$		
	020062	012737	000316	000302		MOV	#316,\$FATAL	:MOVE TO MAILBOX # ***** 316 *****
	020070	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR	
	020072	000000			HALT		:INTERRUPT DID NOT POP CORRECT STATUS	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 746	
1411	020074	000005			2\$:	RESET		
1412	020076	012706	000500		MOV	#BUFF,R6	:STACK SET UP	
1413	020102	012737	020126	000064	MOV	#3\$,64	:INTERRUPT VECTOR	
1414	020110	005037	000066		CLR	66	:CLR NEW STATUS	
1415	020114	012737	000157	177776	MOV	#157,STATUS	:PROCESSOR STATUS	
1416	020122	005137	177564		COM	TTCSR	:SET INTERRUPT ENABLE	
1417	020126	005737	177776		3\$:	TST	STATUS	
1418	020132	001405			BEQ	5\$		
	020134				4\$:			
	020134	012737	000317	000302		MOV	#317,\$FATAL	:MOVE TO MAILBOX # ***** 317 *****
	020142	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
	020144	000000			HALT		:INCORRECT STATUS,OR WRONG \$STNM	
							:TO SCOPE REPLACE HALT WITH 240	
							:AND REPLACE NEXT INST WITH 721	
1419	020146	005037	177564		5\$:	CLR	TTCSR	
1420	020152				6\$:			

1421

.SBTTL TEST #112 - TEST THE 'WAIT' INSTRUCTION

 :TEST 112 - TEST THE 'WAIT' INSTRUCTION

020152	032737	000001	177766	TST112:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
020160	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
020162	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
020170	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
020172	000000				HALT		:CPU POWER BIT FOUND SET
020174	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
020202	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
020206	022737	000112	000304		CMP	#112,\$TESTN	:SEQUENCE ERROR?
020214	001060				BNE	6\$:BRANCH TO ERROR HALT ON SEQ ERROR
1422	020216	005737	016720		TST	PROFTE	
1423	020222	001064			BNE	8\$	
1424	020224	042737	000100	177564	BIC	#100,TPS	:CLEAR INTERRUPT ENABLE
1425	020232	012706	000500		MOV	#BUFF,SP	:SET UP THE STACK
1426	020236	012737	020326	000064	MOV	#4\$,64	:SET UP THE INTERRUPT VECTOR
1427	020244	005037	000066		CLR	66	
1428	020250	105737	177564	1\$:	TSTB	TPS	:WAIT FOR READY
1429	020254	100375			BPL	1\$:TO BE UP
1430	020256	012737	000015	177566	MOV	#15,TPB	:DO A CARRIAGE RETURN
1431	020264	105737	177564	2\$:	TSTB	TPS	:WAIT FOR READY TO COME UP
1432	020270	100375			BPL	2\$	
1433	020272	012737	000015	177566	MOV	#15,TPB	:DO ANOTHER CARRIAGE RETURN
1434	020300	052737	000100	177564	BIS	#100,TPS	:SET THE INTERRUPT ENABLE
1435	020306	005037	177776		CLR	STATUS	:CLEAR THE PSW
1436	020312	000001		3\$:	WAIT		:WAIT FOR THE INTERRUPT
1437	020314	012737	000320	000302	MOV	#320,\$FATAL	:MOVE TO MAILBOX # ***** 320 *****
	020322	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	020324	000000			HALT		:WAIT INSTRUCTION DID NOT LOOP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 733
							:IS THE PSW CORRECT?
1438	020326	005737	177776	4\$:	TST	STATUS	
1439	020332	001405			BEQ	5\$	
	020334	012737	000321	000302	MOV	#321,\$FATAL	:MOVE TO MAILBOX # ***** 321 *****
	020342	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	020344	000000			HALT		:NEW PSW SHOULD HAVE BEEN ZERO
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 723
							:IS THE OLD PC SAVED
1440	020346	023727	000474	020314	5\$:	CMP	BUFF-4,#3\$+2
1441	020354	001405			BEQ	7\$	
	020356				6\$:		
	020356	012737	000322	000302	MOV	#322,\$FATAL	:MOVE TO MAILBOX # ***** 322 *****
	020364	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	020366	000000			HALT		:OLD PC WAS NOT SAVED OR WRONG \$TESTN
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 712
1442	020370	005037	177564	7\$:	CLR	TPS	:CLEAR INTERRUPT ENABLE
1443	020374			8\$:			

1444

.SBTTL TEST #113 - TEST THAT ALL RESERVED INS TRAP

.....
 :TEST 113 - TEST THAT ALL RESERVED INS TRAP

020374	032737	000001	177766	TST113:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
020402	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
020404	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
020412	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
020414	000000				HALT		:CPU POWER BIT FOUND SET
020416	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
020424	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
020430	022737	000113	000304		CMP	#113,\$TESTN	:SEQUENCE ERROR?
020436	001154				BNE	RET4	:BRANCH TO ERROR HALT ON SEQ ERROR
1445	020440	012737	020466		MOV	#TRP244,244	: SET UP TO SEE IF
1446	020446	013737	000010		MOV	10,TENSAV	: THIS PROCESSOR HAS THE
1447	020454	012737	020476		MOV	#TRAP10,10	: FLOATING POINT OPTION
1448	020462	170007			.WORD	170007	: AN ILLEGAL FPP INSTRUCTION
1449	020464	000415			BR	TSFCIS	
1450	020466	013737	021120	021124	TRP244:	MOV	FPP,FINISH
1451	020474	000002			RTI		:IF FPP IN-- RESET END OF TABLE POINTER
1452	020476				TRAP10:		: AND RETURN
1453	020476	005737	000306		TST	\$PASS	: LEAVE THE TABLE ALONE
1454	020502	001003			BNE	1\$:FIRST PASS??
1455	020504	004737	023510		JSR	PC,PRMSG	:BRANCH IF NO
1456	020510	023330			.WORD	MSGNFP	:PRINT MESSAGE POINTED TO BY NEXT WORD
1457	020512	000002			RTI		:USING THIS ADDRESS AS THE START OF THE MESSAGE
1458	020514	000000		1\$:			: RETURN
1459	020516	000000		TENSAV:	.WORD	0	: A PLACE TO STORE CONTENTS OF 10
1460	020520			SAVSTK:	.WORD	0	:LOCATION TO SAVE STACK POINTER
1461	020520	012737	020606	000004	TSFCIS:		:SEE IF PROCESSOR HAS CIS OPTION
1462					MOV	#AROUND,4	:SET TIME OUT TRAP VECTOR
1463	020526	012737	020560	000010	MOV	#TNCIS,10	:SET UP RESERVE INST TRAP VECTOR
1464	020534	012700	160000		MOV	#160000,R0	:POINT R0 TO NON-EXISTED MEMORY LOC.
1465	020540	010637	020516		MOV	SP,SAVSTK	:MOVE STACK POINTER VALUE TO SAVSTK AND
1466	020544	162737	000004	020516	SUB	#4,SAVSTK	:CORRECT TO SHOW 2 EXPECTED PUSHES
1467	020552	076020			.WORD	76020	:CIS INST = L2DR (DESTROYS CONTENTS OF R0,R1,R2,R3)
1468	020554	000000			HALT		:CIS INST FAILED TO TRAP
1469	020556	000430			BR	ADJNC	
1470	020560				TNCIS:		:NO CIS OPTION,EXPECTED TRAP EITHER TO 4 OR 10 DID NOT HAPPEN
1471							
1472	020560	005737	000306		TST	\$PASS	:FIRST PASS
1473	020564	001003			BNE	1\$	
1474	020566	004737	023510		JSR	PC,PRMSG	:PRINT MESSAGE POINTED TO BY THE NEXT WORD
1475	020572	023373			.WORD	MSGNCIS	:USE THIS ADDRESS TO PRINT THE MESSAGE
1476	020574	012703	021014	1\$:	MOV	#TABLE1,TAB	
1477	020600	062716	000002		ADD	#2,(SP)	:CORRECT POINTER TO AFTER THE HALT
1478	020604	000002			RTI		:RETURN
1479	020606	012703	021044	AROUND:	MOV	#TABLE,TAB	:CIS OPTION PRESENT
1480	020612	020637	020516		CMP	SP,SAVSTK	:SEE IF 2 WORDS WERE PUSHED ON THE STACK
1481	020616	001405			BEQ	1\$:BRANCH IF SO - CIS SWITCH IN NORMAL POSITION
1482	020620	004737	023510		JSR	PC,PRMSG	:GO PRINT THE MESSAGE POINTED TO BY THE NEXT WORD
1483	020624	023072			.WORD	CISMSG	:USE THIS ADDRESS TO PRINT THE MESSAGE
1484	020626	000000			HALT		:HALT, ALLOWING USER TO PUT SWITCH IN NORMAL POSITION
1485	020630	000733			BR	TSFCIS	:GO BACK TO CHECK AGAIN
1486	020632	062716	000002	1\$:	ADD	#2,(SP)	:CORRECT POINTER TO AFTER THE HALT
1487	020636	000002			RTI		:RETURN
1488	020640	012737	000246	000244	ADJNC:	MOV	#246,244


```

1489 020646 013737 020514 000010      MOV    TENS AV,10      ; RESTORE THE ILLEGAL INST. VECTOR
1490 020654 012305      GIN1:  MOV    (TAB)+,FIRST ; FIRST OR CURRENT INSTRUCTION
1491 020656 012301      MOV    (TAB)+,LAST   ; LAST INSTRUCTION OR GROUP
1492 020660 020537 021124      CMP    FIRST,FINISH  ; TESTED ALL
1493 020664 001525      BEQ    GIN3          ; YES BRANCH
1494 020666 010537 021126      MOV    FIRST,INST    ; SET UP INST
1495 020672 005237 021126      GIN2:  INC    INST
1496 020676 012737 020720 000010      MOV    RET,10        ; SET UP RETURN FROM TRAP
1497 020704 012706 000500      MOV    #BUFF,SP     ; SET UP STACK POINTER
1498 020710 005037 177776      CLR   CC            ; CLEAR PRIORITY
1499 020714 000137 021126      JMP   INST          ; EXECUTE RESERVED INSTRUCTION
1500
1501      ; TRAPPING SHOULD SEND YOU HERE
1502 020720 020627 000474      RET:   CMP    SP,#BUFF-4 ; TEST DECREMENT OF SP
1503 020724 001405      BEQ   RET1
1504 020726 012737 000323 000302      MOV    #323,$FATAL   ; MOVE TO MAILBOX # ***** 323 *****
1505 020734 005212      INC   (R2)          ; SET MSGTYP TO FATAL ERROR
1506 020736 000000      HALT                ; WRONG DECREMENT
1507 020740 023727 000474 021130      RET1:  CMP    BUFF-4,#INST+2 ; TO SCOPE REPLACE HALT WITH 240
1508 020746 001405      BEQ   RET2          ; AND REPLACE NEXT INST WITH 637
1509 020750 012737 000324 000302      MOV    #324,$FATAL   ; LOC OF INST UNINCREMENTED
1510 020756 005212      INC   (R2)
1511 020760 000000      HALT
1512 020762 005737 000476      RET2:  TST   BUFF-2
1513 020766 001405      BEQ   RET3
1514 020770 020770 000325 000302      RET4:  MOV    #325,$FATAL ; MOVE TO MAILBOX # ***** 325 *****
1515 020776 005212      INC   (R2)          ; SET MSGTYP TO FATAL ERROR
1516 021000 000000      HALT                ; CONDITION CODES SET ON TRAP OR WRONG $STNM
1517 021002 023701 021126      RET3:  CMP    INST, LAST ; TO SCOPE REPLACE HALT WITH 240
1518 021006 001722      BEQ   GIN1          ; AND REPLACE NEXT INST WITH 616
1519 021010 000137 020672      JMP   GIN2
1520 021014 076017      TABLE1: 76017          ; SET UP NEW GROUP
1521 021016 076032      76032          ; FINISH OLD GROUP
1522 021020 076037      76037
1523 021022 076045      76045
1524 021024 076047      76047
1525 021026 076077      76077
1526 021030 076117      76117
1527 021032 076132      76132
1528 021034 076137      76137
1529 021036 076145      76145
1530 021040 076147      76147
1531 021042 076177      76177
1532 021044 000007      TABLE: 7
1533 021046 000077      77
1534 021050 000207      207
1535 021052 000227      227
1536 021054 007077      7077
1537 021056 007777      7777
; RTS,RT1,JMP

```


1532	021060	075037	075037
1533	021062	076017	76017
1534	021064	076032	76032
1535	021066	076037	76037
1536	021070	076045	76045
1537	021072	076047	76047
1538	021074	076132	76132
1539	021076	076137	76137
1540	021100	076145	76145
1541	021102	076147	76147
1542	021104	076077	76077
1543	021106	076117	76117
1544	021110	106377	106377
1545	021112	106477	106477
1546	021114	106677	106677
1547	021116	107777	107777
1548	021120	167777	167777
1549	021122	177777	177777
1550	021124	021124	
1551	021126	000000	
1552	021130	000000	
1553	021132	000000	
1554	021134	000000	
1555	021136	000000	
1556			
1557	021140		

FPP: 167777
177777

FINISH: .
INST: HALT
HALT
HALT
HALT
HALT

GIN3:

; START OF THE FPP INSTRUCTIONS

;END FLAG
;WILL CONTINUE RESERVED INST
;SHOULD TRAP TO LOC 10
;LOC 10 SHOULD SEND YOU TO
;RET

1558

.SBTTL TEST #114 - TEST ILLEGAL HALT

 :TEST 114 - TEST ILLEGAL HALT

021140	032737	000001	177766	TST114:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
021146	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
021150	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
021156	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021160	000000				HALT		:CPU POWER BIT FOUND SET
021162	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
021170	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
021174	022737	000114	000304		CMP	#114,\$TESTN	:SEQUENCE ERROR?
021202	001073				BNE	4\$:BRANCH TO ERROR HALT ON SEQ ERROR
1559 021204	012706	000500			MOV	#BUFF,SP	:STACK POINTER SETUP
1560 021210	005037	177766			CLR	CPUERR	:CLEAR CPU ERROR REGISTER
1561 021214	012737	021250	000004		MOV	#1\$,RTRAP5	:SETUP TRAP RETURN
1562 021222	052737	040000	177776		BIS	#040000,PSW	:GO TO SUPER MODE
1563 021230	000000				HALT		:EXECUTE INST UNDER TEST
1564							
1565 021232	105037	177777			:FAILURE, NO TRAP		
1566 021236	012737	000326	000302		CLRB	PSW+1	:GO BACK TO KERNEL
021244	005212				MOV	#326,\$FATAL	:MOVE TO MAILBOX # ***** 326 *****
021246	000000				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
1567 021250				1\$:			
1568 021250	013737	177766	000502		MOV	CPUERR,RCPUER	:READ AND SAVE CPU ERROR REGISTER
1569 021256	042737	177413	000502		BIC	#CERMSK,RCPUER	:MASK OFF UNUSED CPU ERR REG BITS
1570 021264	022737	000200	000502		CMP	#200,RCPUER	:IS ILLEGAL HALT BIT SET?
1571 021272	001405				BEQ	2\$	
021274	012737	000327	000302		MOV	#327,\$FATAL	:MOVE TO MAILBOX # ***** 327 *****
021302	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021304	000000				HALT		:INCORRECT CPU ERR REG CONTENTS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 736
1572 021306	005037	177766		2\$:	CLR	CPUERR	:CLEAR CPU ERR REG
1573 021312	012737	021346	000004		MOV	#3\$,RTRAP5	:SETUP TRAP RETURN
1574 021320	052737	140000	177776		BIS	#140000,PSW	:GO TO USER MODE
1575 021326	000000				HALT		
1576					:FAILURE, NO TRAP		
1577 021330	105037	177777			CLRB	PSW+1	:GO BACK TO KERNEL
1578 021334	012737	000330	000302		MOV	#330,\$FATAL	:MOVE TO MAILBOX # ***** 330 *****
021342	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021344	000000				HALT		:HALT IN USER MODE FAILED TO TRAP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 716
1579 021346				3\$:			
1580 021346	013737	177766	000502		MOV	CPUERR,RCPUER	:SAVE CPU ERROR REGISTER
1581 021354	042737	177413	000502		BIC	#CERMSK,RCPUER	:MASK OFF UNUSED CPU ERR REG BITS
1582 021362	022737	000200	000502		CMP	#200,RCPUER	:IS ILLEGAL HALT BIT SET?
1583 021370	001405				BEQ	5\$	
021372				4\$:			
021372	012737	000331	000302		MOV	#331,\$FATAL	:MOVE TO MAILBOX # ***** 331 *****
021400	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021402	000000				HALT		:INCORRECT CPU ERR REG CONTENTS
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 677

1584 021404
1585 021404 005037 177766
1586 021410 105037 177777

5\$:

CLR CPUERR
CLRB PSW+1

;GO BACK TO KERNEL MODE

1587

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

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

021414	032737	000001	177766	TST115:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
021422	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
021424	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
021432	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021434	000000				HALT		:CPU POWER BIT FOUND SET
021436	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
021444	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
021450	022737	000115	000304		CMP	#115,\$TESTN	:SEQUENCE ERROR?
021456	001125				BNE	SEQ	:BRANCH TO ERROR HALT ON SEQ ERROR
1588	021460	012706	000500		MOV	#BUFF,SP	:SETUP STACK
1589	021464	052737	040000	177776	BIS	#040000,PSW	:GO TO SUPER MODE
1590	021472	000277			SCC		:SET CC
1591	021474	000231			SPL	1	:SPL SHOULD=NOP IN USER/SUPER MODES
1592	021476	000232			SPL	2	
1593	021500	000233			SPL	3	
1594	021502	000234			SPL	4	
1595	021504	000235			SPL	5	
1596	021506	000236			SPL	6	
1597	021510	000237			SPL	7	
1598	021512	013737	177776	022426	MOV	PSW,SPSW	:SAVE PSW
1599	021520	023727	022426	040017	CMP	SPSW,#040017	:VARIIFY THAT PSW HAS NOT CHANGED
1600	021526	001407			BEQ	1\$	
1601	021530	105037	177777		CLRB	PSW+1	:GO BACK TO KERNEL
1602	021534	012737	000332	000302	MOV	#332,\$FATAL	:MOVE TO MAILBOX # ***** 332 *****
	021542	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	021544	000000			HALT		:PRIORITY LEVELS CHANGE
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 744
1603	021546	012737	040340	177776	1\$:	MOV	#040340,PSW
1604	021554	000257			CCC		:CLEAR CONDITION CODES
1605	021556	000230			SPL	0	:SPL SHOULD=NOP IN SUPERVISOR MODE
1606	021560	013737	177776	022426	MOV	PSW,SPSW	:SAVE PSW
1607	021566	023727	022426	040340	CMP	SPSW,#040340	:VARIIFY THAT PSW PRIORITY AND CONDITION CODES HAVE NOT CHANGE
1608	021574	001407			BEQ	2\$	
1609	021576	105037	177777		CLRB	PSW+1	:GO BACK TO KERNEL
1610	021602	012737	000333	000302	MOV	#333,\$FATAL	:MOVE TO MAILBOX # ***** 333 *****
	021610	005212			INC	(R2)	:SET MSGTYP TO FATAL ERROR
	021612	000000			HALT		:SPL INSTRUCTION CHANGED PSW SHOULD BE NOP
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 721
1611	021614	012737	140000	177776	2\$:	MOV	#140000,PSW
1612	021622	000277			SCC		:SET CC
1613	021624	000231			SPL	1	:SPL SHOULD=NOP IN USER MODE
1614	021626	000232			SPL	2	
1615	021630	000233			SPL	3	
1616	021632	000234			SPL	4	
1617	021634	000235			SPL	5	
1618	021636	000236			SPL	6	
1619	021640	000237			SPL	7	
1620	021642	013737	177776	022426	MOV	PSW,SPSW	:SAVE PSW
1621	021650	023727	022426	140017	CMP	SPSW,#140017	:VARIIFY THAT PSW HAS NOT CHANGED
1622	021656	001407			BEQ	3\$	
1623	021660	105037	177777		CLRB	PSW+1	:GO BACK TO KERNEL

1624	021664	012737	000334	000302		MOV #334,\$FATAL	:MOVE TO MAILBOX # ***** 334 *****
	021672	005212				INC (R2)	:SET MSGTYP TO FATAL ERROR
	021674	000000				HALT	:PRIORITY LEVELS HAS CHANGED
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 670
1625	021676	012737	140340	177776	3\$:	MOV #140340,PSW	:SET PRIORITY TO 7
1626	021704	000257				CCC	:CLEAR CONDITION CODES
1627	021706	000230				SPL 0	:SPL SHOULD=NOP IN USER MODE
1628	021710	013737	177776	022426		MOV PSW,SPSW	:SAVE PSW
1629	021716	023727	022426	140340		CMP SPSW,#140340	:VARIFY THAT PSW PRIORITY AND CONDITION CODES HAVE NOT CHANGE
1630	021724	001407				BEQ FSPL	
1631	021726	105037	177777			CLRB PSW+1	:GO BACK TO KERNEL
1632	021732				SEQ:		
	021732	012737	000335	000302		MOV #335,\$FATAL	:MOVE TO MAILBOX # ***** 335 *****
	021740	005212				INC (R2)	:SET MSGTYP TO FATAL ERROR
	021742	000000				HALT	:SPL INST.CHANGED PSW OR WRONG TEST#
							:TO SCOPE REPLACE HALT WITH 240
							:AND REPLACE NEXT INST WITH 645
1633	021744				F SPL:		
1634	021744	105037	177777			CLRB PSW+1	:GO BACK TO KERNEL
1635							:THIS TEST VARIFIES FOR ALL COMBINATIONS OF PIR AND PROCESSOR
1636							:PRIORITY LEVELS THAT REQUESTS ARE GRANTED (TRAP TO 240 OCCURS
1637							:BY THE PROCESSOR,ONLY WHEN THE PIR IS AT A HIGHER LEVEL THAN
1638							:THE PROCESSOR.
1639							:THE CONTENTS OF SPIR,SPSW AND TRP240 SHOULD BE EXAMINED ON ERROR.
1640							:SPIR BITS 2-0 CONTAINS ONE LESS THAN THE PIR REQUEST LEVEL AT
1641							:THE TIME OF ERROR.SPSW BITS 2-0 CONTAINS THE PROCESSOR PRIORITY
1642							:AT THE TIME OF ERROR.TRP240 INDICATES WHETHER OR NOT A TRAP WAS
1643							:EXPECTED (1= EXPECTING TRAP TO 240)
1644							:THE SPL INSTRUCTION IS USED TO SETUP PROCESSOR PRIORITY.
1645							:NOTE: THIS IS THE FIRST REAL TEST OF THE SPL INST.
1646							:ON ERROR,IF EXPECTED PIRQ TRAP DID NOT OCCURE VARIFY SPL
1647							:OPERATION BY COMPARING SPSW BITS<2-0> WITH PSW PRIORITY
1648							:BITS<7-5>.ON ERROR IF AN UNSPECTED PIRQ TRAP OCCURED
1649							:VARIFY SPL OPERATION BY COMPARING SPSW BITS<2-0> WITH
1650							:PROCESSOR PSW<7-5> ON STACK.
1651							

1652

.SBTTL TEST #116 - TEST PIRQ LEVELS AND SPL INSTRUCTION

 :TEST 116 - TEST PIRQ LEVELS AND SPL INSTRUCTION

021750	032737	000001	177766	TST116:	BIT	#1,CPUERR	:SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
021756	001410				BEQ	100\$:BRANCH AROUND CLEAR AND HALT IF CLEAR
021760	012737	000177	000302		MOV	#177,\$FATAL	:MOVE SPECIAL POWER MONITOR FATAL # TO \$FATAL
021766	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
021770	000000				HALT		:CPU POWER BIT FOUND SET
021772	042737	000001	177766		BIC	#1,CPUERR	:CLEAR THE BIT FOUND SET
022000	005237	000304		100\$:	INC	\$TESTN	:UPDATE TEST NUMBER
022004	022737	000116	000304		CMP	#116,\$TESTN	:SEQUENCE ERROR?
022012	001155				BNE	PTRP	:BRANCH TO ERROR HALT ON SEQ ERROR
1653	022014	005037	022430		CLR	PIRPSW	
1654	022020	012737	022362	000000	MOV	#ZTRP,0	:SET LOCATION ZERO TRAP VEC
1655	022026	012737	000340	000002	MOV	#340,2	
1656	022034	012737	022376	000004	MOV	#T4TRP,4	:SET UP FAILURE TRAP VEC
1657	022042	012737	000340	000006	MOV	#340,6	
1658	022050	012737	022412	000024	MOV	#T24TRP,24	:SETUP POWER FAIL VEC
1659	022056	012737	000340	000026	MOV	#340,26	
1660	022064	012737	022324	000240	MOV	#PQTRP,PIRVC1	:SETUP PIRQ VEC
1661	022072	012737	000340	000242	MOV	#340,PIRVC2	:SET 242 TO PRIORITY 7
1667	022100	012706	000500		MOV	#BUFF,SP	:SETUP STACK
1668	022104	052737	000340	177776	MLOOP:	BIS	#340,PSW
1669							:SET PROCESSOR PRIORITY TO 7
1670	022112	013700	022430				: COMPUTE EXPECTED RESULT
1671	022116	042700	177707		MOV	PIRPSW,R0	
1672	022122	006200			BIC	#177707,R0	:SETUP SPIR
1673	022124	006200			ASR	R0	
1674	022126	006200			ASR	R0	
1675	022130	010037	022424		ASR	R0	
1676	022134	013700	022430		MOV	R0,SPIR	:SAVE R0 IN SPIR
1677	022140	042700	177770		MOV	PIRPSW,R0	
1678	022144	010037	022426		BIC	#177770,R0	:SETUP SPSW
1679	022150	023737	022424	022426	MOV	R0,SPSW	
1680	022156	002003			CMP	SPIR,SPSW	
1681	022160	005037	022432		BGE	2\$:BRANCH IF PIR > PSW
1682	022164	000403			CLR	TRP240	:CLEAR FLAG
1683	022166	012737	177777	022432	BR	3\$	
1684	022174	004737	022434		2\$:	MOV	#177777,TRP240
1685					3\$:	JSR	PC,SETPIRQ
1686	022200	042737	000007	022214			:SET FLAG
1687	022206	053737	022426	022214			:SETUP PIRQ BASE ON THE # IN SPIR
1688	022214	000230					:SET PSW PRIORITY BASE ON THE # IN SPSW
1689					BIC	#7,1\$:CLEAR LSB 3 BITS OF SPL INST.AT 1\$
1690	022216	000240			BIS	SPSW,1\$:SET LSB 3 BITS OF SPL INST.TO DESIDED PROC. PRI.
1691	022220	005737	022432		1\$:	SPL	0
1692	022224	001405					:THE ACTUAL PRIORITY SET INTO THE PSW IS CONTROLLED BY
022226	012737	000336	000302				:THE PREVIOUS TWO INSTRUCTION.
022234	005212				NOP		
022236	000000				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
1693	022240	000240			NXTST:	NOP	:SET UP FOR PASS COUNT
1694							
1695							
1696	022242	005237	022430		INC	PIRPSW	:SETUP FOR NEXT PASS THROUGH LOOP

1697	022246	023727	022430	000070		CMP	PIRPSW,#70	
1698	022254	002711				BLT	MLOOP	:BRANCH TO MLOOP IF COUNT IS LESS THAN 70
1699	022256	005037	177772			CLR	PIRQ	:DONE WITH LOOPING PIRQ REQUESTS
1700	022262	012737	000006	000004		MOV	#6,4	:RESTORE RETURNS
1701	022270	005037	000006			CLR	6	
1702	022274	005037	000000			CLR	0	
1703	022300	012737	023424	000024		MOV	#PWRDWN,24	
1704	022306	012737	000242	000240		MOV	#242,240	
1705	022314	005037	000242			CLR	242	
1706	022320	000137	022630			JMP	TST117	
1707	022324	005737	022432		PQTRP:	TST	TRP240	:PRIORITY TRAP SERVICE
1708	022330	001343				BNE	NXTST	
	022332	012737	000337	000302		MOV	#337,\$FATAL	:MOVE TO MAILBOX # ***** 337 *****
	022340	005212				INC	(R2)	:SET MSGTYP TO FATAL ERROR
	022342	000000				HALT		:EXPECTED NO PIRQ TRAP,BUT GOT ONE
								:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 655
1709	022344	000735				BR	NXTST	
1713	022346				PTRP:	MOV	#340,\$FATAL	:MOVE TO MAILBOX # ***** 340 *****
	022346	012737	000340	000302		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	022354	005212				HALT		:WRONG TEST NUMBER
	022356	000000						:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 615
1717	022360	000523				BR	TST117	
1718	022362				ZTRP:	MOV	#341,\$FATAL	:MOVE TO MAILBOX # ***** 341 *****
1719	022362	012737	000341	000302		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	022370	005212				HALT		:UNSPECTED TRAP TO ZERO OCCURED DURING PIRQ TST
	022372	000000						:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 641
1720	022374	000721				BR	NXTST	
1721	022376				T4TRP:	MOV	#342,\$FATAL	:MOVE TO MAILBOX # ***** 342 *****
1722	022376	012737	000342	000302		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	022404	005212				HALT		:UNSPECTED TRAP TO 4 DURING PIRQ TESTING
	022406	000000						:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 633
1723	022410	000713				BR	NXTST	
1724	022412				T24TRP:	MOV	#343,\$FATAL	:MOVE TO MAILBOX # ***** 343 *****
1725	022412	012737	000343	000302		INC	(R2)	:SET MSGTYP TO FATAL ERROR
	022420	005212				HALT		:UNSPECTED POWER FAIL TRAP DURING PIRQ TESTING
	022422	000000						:TO SCOPE REPLACE HALT WITH 240
								:AND REPLACE NEXT INST WITH 625
1726	022424	000000				SPIR:	.WORD	0
1727	022426	000000				SPSW:	.WORD	0
1728	022430	000000				PIRPSW:	.WORD	0

1730 177772 PIRQ=177772
 1731 000240 PIRVC1=240
 1732 000242 PIRVC2=242
 1733 022432 000000 TRP240: .WORD 0
 1734
 1735

1736	022434				000000	SETPIRQ:		
1737	022434	023727	022424				CMP	SPIR,#0
1738	022442	001435					BEQ	1\$
1739	022444	023727	022424		000001		CMP	SPIR,#1
1740	022452	001435					BEQ	2\$
1741	022454	023727	022424		000002		CMP	SPIR,#2
1742	022462	001435					BEQ	3\$
1743	022464	023727	022424		000003		CMP	SPIR,#3
1744	022472	001435					BEQ	4\$
1745	022474	023727	022424		000004		CMP	SPIR,#4
1746	022502	001435					BEQ	5\$
1747	022504	023727	022424		000005		CMP	SPIR,#5
1748	022512	001435					BEQ	6\$
1749	022514	023727	022424		000006		CMP	SPIR,#6
1750	022522	001435					BEQ	7\$
1751	022524	012737	000344		000302		MOV	#344,\$FATAL
	022532	005212					INC	(R2)
	022534	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 560

1752	022536	012737	001000	177772	1\$:	MOV	#1000,PIRQ
1753	022544	000430				BR	10\$
1754	022546	012737	002000	177772	2\$:	MOV	#2000,PIRQ
1755	022554	000424				BR	10\$
1756	022556	012737	004000	177772	3\$:	MOV	#4000,PIRQ
1757	022564	000420				BR	10\$
1758	022566	012737	010000	177772	4\$:	MOV	#10000,PIRQ
1759	022574	000414				BR	10\$
1760	022576	012737	020000	177772	5\$:	MOV	#20000,PIRQ
1761	022604	000410				BR	10\$
1762	022606	012737	040000	177772	6\$:	MOV	#40000,PIRQ
1763	022614	000404				BR	10\$
1764	022616	012737	100000	177772	7\$:	MOV	#100000,PIRQ
1765	022624	000400				BR	10\$
1766	022626	000207			10\$:	RTS	PC

1767
 1768
 1769
 1770
 1771
 1772
 1773
 1774
 1775
 1776
 1777

:TABLE FOR PIRQ SETUP
 :SPIR PIR LEVEL
 :0000 1
 :0001 2
 :0002 3
 :0003 4
 :0004 5
 :0005 6
 :0006 7

SETPIRQ	# LOADED INTO PIRQ REG RO
BIT9	1000
BIT10	2000
BIT11	4000
BIT12	10000
BIT13	20000
BIT14	40000
BIT15	100000

1778

```

.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: BIT #1, CPUERR ;SEE IF BIT 0 OF CPUERR (POWER MONITOR BIT) IS SET
        BEQ 100$ ;BRANCH AROUND CLEAR AND HALT IF CLEAR
        MOV #177,$FATAL ;MOVE SPECIAL POWER MONITOR FATAL # TO $FATAL
        INC (R2) ;SET MSGTYP TO FATAL ERROR
        HALT ;CPU POWER BIT FOUND SET
        BIC #1, CPUERR ;CLEAR THE BIT FOUND SET
100$: INC $TESTN ;UPDATE TEST NUMBER
      CMP #117,$TESTN ;SEQUENCE ERROR?
      BNE OATSE ;BRANCH 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$: CMP #30340,177776 ;SEE IF PREVIOUS MODE USER AND PRIORITY 7 LOADED
     BEQ END ;BRANCH OUT IF OK
:*****
:IF PSW CONTAINS 30000, YOU MIGHT BE MISSING ECO #6.
:*****
1789 022734 012737 000345 000302 MOV #345,$FATAL ;MOVE TO MAILBOX # ***** 345 *****
     022742 005212 INC (R2) ;SET MSGTYP TO FATAL ERROR
     022744 000000 HALT ;PSW NOT PROPERLY LOADED
     ;TO SCOPE REPLACE HALT WITH 240
     ;AND REPLACE NEXT INST WITH 752
     ;BRANCH TO END ROUTINE
1790 022746 000405 BR END
1791 022750 OATSE: MOV #346,$FATAL ;MOVE TO MAILBOX # ***** 346 *****
     022750 012737 000346 000302 INC (R2) ;SET MSGTYP TO FATAL ERROR
     022756 005212 HALT ;WRONG $STNM
     022760 000000 ;TO SCOPE REPLACE HALT WITH 240
     ;AND REPLACE NEXT INST WITH 744
  
```


1793					.SBTTL	END OF PASS ROUTINE	
1794	022762	005237	000306		END: INC	\$PASS	
1795	022766	105237	023070		INCB	PASSPT	: SHOULD PRINT THIS PASS?
1796	022772	001017			BNE	ACT	: NO
1797	022774	132737	000040	000321	BITB	#40,\$ENV	: WILL APT ALLOW PRINTING?
1798	023002	001013			BNE	ACT	: NO
1799	023004	023727	000042	023042	CMP	42,\$SENDAD	
1800	023012	001407			BEQ	ACT	
1801	023014	004737	023510		JSR	PC,PRMSG	: PRINT MESSAGE POINTED TO BY THE NEXT WORD
1802	023020	023215			.WORD	MSG	: USE THIS ADDRESS TO PRINT THE MESSAGE
1803	023022	000005			RESET		
1804	023024	012737	177761	023070	MOV	#177761,PASSPT	: DO IT ABOUT 15 DECIMAL TIMES
1805	023032	013700	000042		ACT: MOV	42,R0	: CHECK ACT
1806	023036	001405			BEQ	GOAGIN	: KEEP GOING
1807	023040	000005			RESET		
1808	023042	004710			\$SENDAD: JSR	PC,(R0)	: ACT HOOKS
1809	023044	000240			NOP		
1810	023046	000240			NOP		
1811	023050	000240			NOP		
1812	023052	012737	000012	000010	GOAGIN: MOV	#12,10	
1813	023060	005037	000012		CLR	12	
1814	023064	000137	000540		JMP	RESTR	: DO NEXT PASS
1815	023070	177777			PASSPT: -1		

1816					.SBTTL	ASCII MESSAGES
1817	023072	015	012	103	CISMSG: .ASCII	<15><12>.CIS SWITCH IS IN THE MAINTENANCE POSITION.<15><12>
1818	023147	120	125	124	.ASCIZ	.PUT IN NORMAL POSITION AND CONTINUE.<15><12>
1819	023215	015	012	105	MSG: .ASCIZ	<15><12>.END OF CKKABCO 11/44 TRAPS.
1820	023252	015	012	103	PNAME: .ASCIZ	<15><12>+CKKABCO 11/44 TRAPS+
1821	023300	015	012	103	TITLE: .ASCIZ	<15><12>+CKKABCO 11/44 TRAPS+<15><12>
1822	023330	015	012	116	MSGNFP: .ASCIZ	<15><12>.NO FLOATING POINT OPTION PRESENT.
1823	023373	015	012	116	MSGNCIS: .ASCIZ	<15><12>.NO CIS OPTION PRESENT .
1824					.EVEN	


```

1825                                     .SBTTL  POWER DOWN AND POWER UP ROUTINES
1826 023424 012737 023434 000024 PWRDWN: MOV    #PWRUP,24
1827 023432 000000                                     HALT
1828
1829 023434 012737 023424 000024 PWRUP:  MOV    #PWRDWN,24
1830 023442 012706 000500                                     MOV    #BUFF,SP
1831 023446 132737 000040 000321      BITB   #40,SEVM      :WILL APT ALLOW PRINTING?
1832 023454 001003                                     BNE    PFRES       :NO
1833 023456 004737 023510      JSR    PC,PRMSG    :PRINT MESSAGE POINTED TO BY THE NEXT WORD
1834 023462 023470      .WORD  MSGPWF      :USE THIS ADDRESS TO PRINT THE MESSAGE
1835 023464 000137 000540      PFRES: JMP    RESTR
1836 023470      015      012      120  MSGPWF: .ASCIZ <15><12>.POWER FAILED!
    
```


					.SBTTL	SUBROUTINE TO PRINT A MESSAGE	
1837							
1838	023510	017600	000000		PRTMSG: MOV	@0(SP),R0	:MOVE MESSAGE START ADDRESS TO R0
1839	023514	062716	000002		ADD	#2,(SP)	:FUDGE RETURN OVER STARTING ADDRESS PARAMETER
1840	023520	132737	000040	000321	BITB	#40,SENVH	:WILL APT ALLOW PRINTING?
1841	023526	001011			BNE	18	:BRANCH IF NO
1842	023530	105737	177564	28:	TSTB	TPS	:TTY READY
1843	023534	100375			BPL	28	:NO WAIT
1844	023536	112037	177566		MOVB	(R0)+,TPB	:PRINT CHARACTER
1845	023542	001372			BNE	28	:NEXT IF NOT DONE.
1846	023544	105737	177564	38:	TSTB	TPS	
1847	023550	100375			BPL	38	
1848	023552	000207		18:	RTS	PC	

CKKABCO 11/44 TRAPS MACRO M1113 23-APR-81 12:47 PAGE 106
SUBROUTINE TO PRINT A MESSAGE

K 10

SEQUENCE 127-

1849

000001

.END

ABASE = 000000	BUFF = 000500	PTRP = 022346	TST113 = 020374	TST64 = 013550
ACDW1 = 000000	CC = 177776	PWRDWN = 023424	TST114 = 021140	TST65 = 013656
ACDW2 = 000000	CERMSK = 177413	PWRUP = 023434	TST115 = 021414	TST66 = 013764
ACPUOP = 000000	CISMSG = 023072	RCPUER = 000502	TST116 = 021750	TST67 = 014072
ACT = 023032	CPUERR = 177766	RESTRT = 000540	TST117 = 022630	TST7 = 002516
ADDW0 = 000000	DL11W = 016722	RET = 020720	TST12 = 003334	TST70 = 014220
ADDW1 = 000000	DL11W1 = 016726	RET1 = 020740	TST13 = 003440	TST71 = 014326
ADDW10 = 000000	END = 022762	RET2 = 020762	TST14 = 003540	TST72 = 014464
ADDW11 = 000000	FINISH = 021124	RET3 = 021002	TST15 = 003642	TST73 = 014570
ADDW12 = 000000	FIRST = %000005	RET4 = 020770	TST16 = 004016	TST74 = 014714
ADDW13 = 000000	FPP = 021120	RTRAP = 000010	TST17 = 004342	TST75 = 015026
ADDW14 = 000000	FSPL = 021744	RTRAP1 = 000034	TST2 = 001232	TST76 = 015136
ADDW15 = 000000	GIN1 = 020654	RTRAP2 = 000020	TST20 = 004470	TST77 = 015276
ADDW2 = 000000	GIN2 = 020672	RTRAP3 = 000030	TST21 = 004562	TTCSR = 177564
ADDW3 = 000000	GIN3 = 021140	RTRAP4 = 000014	TST22 = 004662	T24TRP = 022412
ADDW4 = 000000	GOAGIN = 023052	RTRAP5 = 000004	TST23 = 004764	T4TRP = 022376
ADDW5 = 000000	HERE = 000000	R6 = %000006	TST24 = 005142	ZTRP = 022362
ADDW6 = 000000	HLT = 000000	R7 = %000007	TST25 = 005500	\$APTHD = 000330
ADDW7 = 000000	ILLA = 004700	SAVSTK = 020516	TST26 = 005572	\$CPUOP = 000326
ADDW8 = 000000	ILLB = 000100	SEQ = 021732	TST27 = 005672	\$DEVCT = 000310
ADDW9 = 000000	INST = 021126	SETPIR = 022434	TST3 = 001612	\$ENDAD = 023042
ADEVCT = 000000	KTSTA = 000506	SPIR = 022424	TST30 = 005774	\$ENV = 000320
ADEVM = 000000	KTVEC = 000504	SPSW = 022426	TST31 = 006152	\$ENVM = 000321
ADJNC = 020640	K1 = 000626	STATUS = 177776	TST32 = 006472	\$ERN = 000347
AENV = 000000	K10 = 000644	TAB = %000003	TST33 = 006620	\$ERROR = 000302
AENVM = 000000	K11 = 000646	TABLE = 021044	TST34 = 006712	\$ETABL = 000320
AFATAL = 000000	K12 = 000650	TABLE1 = 021014	TST35 = 007012	\$ETEND = 000330
AMADR1 = 000000	K2 = 000630	TENSAV = 020514	TST36 = 007114	\$FATAL = 000302
AMADR2 = 000000	K3 = 000632	TITLE = 023300	TST37 = 007272	\$HIBTS = 000330
AMADR3 = 000000	K4 = 000634	TNCIS = 020560	TST4 = 002102	\$MAIL = 000300
AMADR4 = 000000	K5 = 000636	TPB = 177566	TST40 = 007630	\$MBADR = 000332
AMAMS1 = 000000	K6 = 000640	TPS = 177564	TST41 = 007722	\$MSGAD = 000314
AMAMS2 = 000000	K7 = 000642	TRAPA = 000010	TST42 = 010022	\$MSGLG = 000316
AMAMS3 = 000000	LAST = %000001	TRAP10 = 020476	TST43 = 010124	\$MSGTY = 000300
AMAMS4 = 000000	MLOOP = 022100	TRCSR = 177560	TST44 = 010302	\$PASS = 000306
AMSGAD = 000000	MSG = 023215	TRP240 = 022432	TST45 = 010622	\$PASTM = 000336
AMSGLG = 000000	MSGNCI = 023373	TRP244 = 020466	TST46 = 010714	\$SVPC = 000300
AMSGTY = 000000	MSGNFP = 023330	TRT = 000003	TST47 = 011014	\$SWR = 000000
AMTYP1 = 000000	MSGPWF = 023470	TSFCIS = 020520	TST5 = 002324	\$SWREG = 000322
AMTYP2 = 000000	NOP = 000240	TST1 = 000652	TST50 = 011116	\$TESTN = 000304
AMTYP3 = 000000	NXTST = 022240	TST10 = 002620	TST51 = 011274	\$TN = 000120
AMTYP4 = 000000	OATSE = 022750	TST100 = 015422	TST52 = 011612	\$TSTM = 000334
APASS = 000000	PASSPT = 023070	TST101 = 015744	TST53 = 011774	\$TSTNM = 000304
APRIOR = 000000	PFRES = 023464	TST102 = 016114	TST54 = 012076	\$UNIT = 000312
AROUND = 020606	PIRPSW = 022430	TST103 = 016240	TST55 = 012202	\$UNITM = 000340
ASWREG = 000000	PIRQ = 177772	TST104 = 016734	TST56 = 012364	\$USWR = 000324
ATESTN = 000000	PIRVC1 = 000240	TST105 = 017112	TST57 = 012710	\$X = 022674
AUNIT = 000000	PIRVC2 = 000242	TST106 = 017300	TST6 = 002416	\$XX = 177745
AUSWR = 000000	PNAME = 023252	TST107 = 017466	TST60 = 013050	\$XXX = 000744
AVECT1 = 000000	PQTRP = 022324	TST11 = 002776	TST61 = 013150	\$Y = 022014
AVECT2 = 000000	PROFTE = 016720	TST110 = 017622	TST62 = 013334	\$YY = 022100
BEGIN = 000510	PRTMSG = 023510	TST111 = 017746	TST63 = 013442	.\$X = 000330
BELL = 000240	PSW = 177776	TST112 = 020152		

. ABS. 023554 000
000000 001
ERRORS DETECTED: 0

CKKABC0 11/44 TRAPS
SYMBOL TABLE

MACRO M1113 23-APR-81 12:47 PAGE 106-2

M 10

SEQUENCE 129

VIRTUAL MEMORY USED: 8397 WORDS (33 PAGES)
DYNAMIC MEMORY: 9474 WORDS (36 PAGES)
ELAPSED TIME: 00:07:47
CKKABC.BIN,CKKABC/CR/-SP/NL:TOC=CKKABC.MLB/ML,CKKABC.P11

CKKABC

CREATED BY MACRO ON 23-APR-81 AT 12:51

PAGE 1

N 10

SEQUENCE 130
REF V01

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
ABASE	=	000000	17-389
ACDW1	=	000000	17-389
ACDW2	=	000000	17-389
ACPUOP	=	000000	17-389 17-389
ACT	=	023032	102-1796 102-1798 102-1800 102-1805
ADDW0	=	000000	17-389
ADDW1	=	000000	17-389
ADDW10	=	000000	17-389
ADDW11	=	000000	17-389
ADDW12	=	000000	17-389
ADDW13	=	000000	17-389
ADDW14	=	000000	17-389
ADDW15	=	000000	17-389
ADDW2	=	000000	17-389
ADDW3	=	000000	17-389
ADDW4	=	000000	17-389
ADDW5	=	000000	17-389
ADDW6	=	000000	17-389
ADDW7	=	000000	17-389
ADDW8	=	000000	17-389
ADDW9	=	000000	17-389
ADEVCT	=	000000	17-389 17-389
ADEVN	=	000000	17-389
ADJNC	=	020640	96-1469 #96-1488
AENV	=	000000	17-389 17-389
AENVN	=	000000	17-389 17-389
AFATAL	=	000000	17-389 17-389
AMADR1	=	000000	17-389
AMADR2	=	000000	17-389
AMADR3	=	000000	17-389
AMADR4	=	000000	17-389
AMAMS1	=	000000	17-389
AMAMS2	=	000000	17-389
AMAMS3	=	000000	17-389
AMAMS4	=	000000	17-389
AMSGAD	=	000000	17-389 17-389
AMSGLG	=	000000	17-389 17-389
AMSGTY	=	000000	17-389 17-389
AMTYP1	=	000000	17-389
AMTYP2	=	000000	17-389
AMTYP3	=	000000	17-389
AMTYP4	=	000000	17-389
APASS	=	000000	17-389 17-389
APRIOR	=	000000	17-389
AROUND	=	020606	96-1461 #96-1479
ASWREG	=	000000	17-389 17-389
ATESTN	=	000000	17-389 17-389
AUNIT	=	000000	17-389 17-389
AUSWR	=	000000	17-389 17-389
AVECT1	=	000000	17-389
AVECT2	=	000000	17-389
BEGIN	=	000510	16-382 16-385 #20-401

SYMBOL CROSS REFERENCE

SYMBOL VALUE

BELL = 000240
 BUFF 000500

REFERENCES

#14-359									
#19-397	26-560	27-566	27-569	28-572	28-575	29-578	29-583	29-585	
29-590	30-593	30-604	31-621	32-627	32-630	33-633	33-636	34-639	
34-644	34-646	34-650	35-653	35-664	36-680	37-689	38-695	38-698	
39-701	39-704	40-707	40-712	40-714	40-719	41-722	41-733	42-748	
43-754	43-757	44-760	44-763	45-766	45-771	45-773	45-778	46-781	
46-792	47-806	48-816	49-822	49-825	50-828	50-831	51-834	51-839	
51-842	51-847	52-850	52-861	53-883	54-889	54-892	55-895	55-898	
56-901	56-906	56-908	56-913	57-916	57-927	58-939	59-945	59-948	
60-951	60-954	61-958	61-963	61-965	61-970	62-973	62-984	63-1000	
64-1012	64-1015	65-1018	65-1021	66-1024	66-1029	66-1031	66-1036	67-1039	
67-1050	79-1133	80-1142	80-1149	81-1152	81-1157	82-1161	83-1172	84-1188	
85-1200	85-1204	85-1206	85-1210	85-1212	85-1216	85-1218	85-1224	86-1227	
87-1241	87-1247	88-1264	88-1286	88-1310	88-1317	90-1343	91-1360	93-1388	
94-1404	94-1412	95-1425	95-1440	96-1497	96-1502	96-1505	96-1508	97-1559	
98-1588	99-1667	104-1830							

CC = 177776

#14-366	*29-580	*29-587	30-602	30-612	*34-641	*34-648	35-662	35-673	
*40-709	*40-716	41-731	41-741	*45-768	*45-775	46-790	46-801	*51-836	
*51-844	52-859	52-869	*56-903	*56-910	57-925	57-935	*61-960	*61-967	
62-982	62-992	*66-1026	*66-1033	67-1048	67-1058	*96-1498			
#14-363	63-1006	68-1069	88-1272	88-1300	97-1569	97-1581			

CERMSK = 177413
 CISMSG 023072
 CPUERR = 177766

#14-367	22-432	*22-432	23-482	*23-482	24-522	*24-522	25-545	*25-545	
26-559	*26-559	27-565	*27-565	28-571	*28-571	29-577	*29-577	30-592	
*30-592	31-618	*31-618	32-626	*32-626	33-632	*33-632	34-638	*34-638	
35-652	*35-652	36-677	*36-677	37-688	*37-688	38-694	*38-694	39-700	
*39-700	40-706	*40-706	41-721	*41-721	42-747	*42-747	43-753	*43-753	
44-759	*44-759	45-765	*45-765	46-780	*46-780	47-803	*47-803	48-815	
*48-815	49-821	*49-821	50-827	*50-827	51-833	*51-833	52-849	*52-849	
53-882	*53-882	54-888	*54-888	55-894	*55-894	56-900	*56-900	57-915	
*57-915	58-938	*58-938	59-944	*59-944	60-950	*60-950	61-957	*61-957	
62-972	*62-972	63-996	*63-996	*63-997	63-998	63-1005	*63-1009	64-1011	
*64-1011	65-1017	*65-1017	66-1023	*66-1023	67-1038	*67-1038	68-1061	*68-1061	
*68-1062	68-1068	*68-1072	69-1073	*69-1073	70-1079	*70-1079	71-1106	*71-1106	
72-1107	*72-1107	73-1108	*73-1108	74-1109	*74-1109	75-1110	*75-1110	76-1111	
*76-1111	77-1114	*77-1114	78-1116	*78-1116	79-1132	*79-1132	80-1141	*80-1141	
81-1151	*81-1151	82-1160	*82-1160	83-1170	*83-1170	84-1187	*84-1187	85-1199	
*85-1199	86-1226	*86-1226	87-1240	*87-1240	88-1254	*88-1254	*88-1261	88-1271	
*88-1284	88-1299	89-1324	*89-1324	90-1340	*90-1340	91-1357	*91-1357	92-1376	
*92-1376	93-1387	*93-1387	94-1400	*94-1400	95-1421	*95-1421	96-1444	*96-1444	
97-1558	*97-1558	*97-1560	97-1568	*97-1572	97-1580	*97-1585	98-1587	*98-1587	
99-1652	*99-1652	101-1778	*101-1778						

DL11W 016722
 DL11W1 016726
 END 022762
 FINISH 021124
 FIRST =%000005
 FPP 021120
 FSPL 021744
 GIN1 020654
 GIN2 020672
 GIN3 021140

88-1318	#88-1322								
88-1320	#88-1323								
101-1785	101-1790	#102-1794							
*96-1450	96-1492	#96-1550							
#14-343	*96-1490	96-1492	96-1494						
96-1450	#96-1548								
98-1630	#98-1633								
#96-1490	96-1511								
#96-1495	96-1512								
96-1493	#96-1557								

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES
GOAGIN	023052	102-1806 #102-1812
HERE	= 000000	#21-420 *22-434 *22-439 22-479
HLT	= 000000	#14-348
ILLA	= 004700	#14-364 76-1111
ILLB	= 000100	#14-365 77-1114
INST	021126	*96-1494 *96-1495 96-1499 96-1505 96-1510 #96-1551
KTSTA	000506	#19-400
KTVEC	000504	#19-399
K1	000626	#21-422 *23-484 23-485 23-488 *23-492 23-493 23-496 *23-499 23-501
		*23-507 23-509 *23-515 23-517 23-520
K10	000644	#21-429 24-529 24-532 24-534 24-537 24-540 24-543
K11	000646	#21-430
K12	000650	#21-431
K2	000630	#21-423
K3	000632	#21-424
K4	000634	#21-425
K5	000636	#21-426 *23-483 23-486 *23-491 *23-500 23-502 23-504 *23-508 23-510
		23-512 *23-516 23-518
K6	000640	#21-427 23-494 24-532
K7	000642	#21-428 24-523 24-523 24-526 24-526 24-529 24-534 24-543
LAST	=%000001	#14-337 *96-1491 96-1510
MLOOP	022100	#99-1667 99-1698
MSG	023215	102-1802 #103-1819
MSGNCI	023373	96-1475 #103-1823
MSGNFP	023330	96-1456 #103-1822
MSGPWF	023470	104-1834 #104-1836
NOP	= 000240	#14-360
NXTST	022240	99-1692 #99-1693 99-1708 99-1709 99-1720 99-1723
OATSE	022750	101-1778 #101-1791
PASSPT	023070	*20-402 *102-1795 *102-1804 #102-1815
PFRES	023464	104-1832 #104-1835
PIRPSW	022430	*99-1653 99-1670 99-1676 *99-1696 99-1697 #99-1728
PIRQ	= 177772	*99-1699 #100-1730 *100-1752 *100-1754 *100-1756 *100-1758 *100-1760 *100-1762 *100-1764
PIRVC1	= 000240	*99-1660 #100-1731
PIRVC2	= 000242	*99-1661 #100-1732
PNAME	023252	#103-1820
PQTRP	022324	99-1660 #99-1707
PROFTE	016720	*88-1316 #88-1321 *88-1322 89-1325 90-1341 91-1358 92-1377 94-1401 95-1422
PRTMSG	023510	20-405 96-1455 96-1474 96-1482 102-1801 104-1833 #105-1838
PSW	= 177776	#14-369 *97-1562 *97-1565 *97-1574 *97-1577 *97-1586 *98-1589 98-1598 *98-1601
		*98-1603 98-1606 *98-1609 *98-1611 98-1620 *98-1623 *98-1625 98-1628 *98-1631
		*98-1634 *99-1668
PTRP	022346	99-1652 #99-1713
PWRDWN	023424	20-409 99-1703 #104-1826 104-1829
PWRUP	023434	104-1826 #104-1829
RCPUER	000502	#19-398 *63-1006 *63-1007 *68-1068 *68-1069 68-1070 *88-1271 *88-1272
		88-1277 88-1280 *88-1299 *88-1300 88-1305 88-1308 *97-1568 *97-1569 97-1570
		*97-1580 *97-1581 97-1582
RESTRT	000540	20-404 #20-407 102-1814 104-1835
RET	020720	96-1496 #96-1502
RET1	020740	96-1503 #96-1505
RET2	020762	96-1506 #96-1508

SYMBOL CROSS REFERENCE

SYMBOL	CROSS REFERENCE	SYMBOL	VALUE	REFERENCES
RET3	021002			96-1509 #96-1510
RET4	020770			96-1444 #96-1509
RTCAP	= 000010			#14-363 *26-561 *27-567 *28-573 *29-579 *29-586 *30-594 *30-595 *30-605
				*30-606 *53-884 *54-890 *55-896 *56-902 *56-909 *57-917 *57-918 *57-928
				*57-929 *58-940 *59-946 *60-952 *61-959 *61-966 *62-974 *62-975 *62-985
				*62-986
RTRAP1	= 000034			#14-354 *31-622 *32-628 *33-634 *34-640 *34-647 *35-654 *35-655 *35-665
				*35-666
RTRAP2	= 000020			#14-353 *37-690 *38-696 *39-702 *40-708 *40-715 *41-723 *41-724 *41-734
				*41-735
RTRAP3	= 000030			#14-352 *42-749 *43-755 *44-761 *45-767 *45-774 *46-782 *46-783 *46-793
				*46-794
RTRAP4	= 000014			#14-351 *48-817 *49-823 *50-829 *51-835 *51-843 *52-851 *52-852 *52-862
				*52-863 *79-1134 *80-1143 *81-1153
RTRAP5	= 000004			#14-350 *63-1001 *64-1013 *65-1019 *66-1025 *66-1032 *67-1040 *67-1041 *67-1051
				*67-1052 *97-1561 *97-1573
R6	=%000006			#14-344 *20-408 *22-433 *22-434 22-435 *22-438 *22-439 22-440 *22-443
				*22-444 *22-444 22-445 *22-448 *22-450 22-451 *22-454 *22-456 22-457
				*22-460 *22-462 *22-466 *22-468 22-469 *22-472 *22-474 *22-478 *22-479
				22-480 *23-486 *23-487 *23-494 *23-495 *23-502 *23-503 *23-510 *23-511
				*23-518 *23-519 *68-1063 *68-1065 *69-1074 *69-1076 69-1077 *70-1080 *70-1087
				*70-1089 *70-1091 *70-1093 *71-1106 *72-1107 *73-1108 *74-1109 *75-1110 *76-1111
				76-1112 *77-1114 *78-1119 *78-1120 *78-1121 *78-1122 *78-1123 *78-1124 *78-1125
				*78-1126 *78-1127 *78-1128 *85-1200 *85-1206 *85-1212 *85-1218 *86-1227 *87-1241
				*89-1329 *90-1343 *91-1360 *93-1388 *93-1390 *93-1391 *94-1404 *94-1412
				#14-346 *85-1202 *85-1208 *85-1214 *85-1221
R7	=%000007			#96-1459 *96-1465 *96-1466 96-1480
SAVSTK	020516			98-1587 #98-1632
SEQ	021732			99-1684 #100-1736
SETPIR	022434			*99-1675 99-1679 #99-1726 100-1737 100-1739 100-1741 100-1743 100-1745 100-1747
SPIR	022424			100-1749
				*98-1598 98-1599 *98-1606 98-1607 *98-1620 98-1621 *98-1628 98-1629 *99-1678
SPSW	022426			99-1679 #99-1727
				#14-361 *25-547 *25-553 88-1296 *89-1328 *89-1333 *90-1344 *90-1350 *91-1361
				*93-1396 *94-1406 94-1409 *94-1415 94-1417 *95-1435 95-1438
				#14-340 *96-1476 *96-1479
TAB	=%000003			*96-1479 #96-1526
TABLE	021044			96-1476 #96-1514
TABLE1	021014			*96-1446 #96-1458 96-1489
TENSAV	020514			20-406 #103-1821
TITLE	023300			96-1463 #96-1470
TNCIS	020560			#14-358 *95-1430 *95-1433 *105-1844
TPB	= 177566			#14-357 88-1319 *95-1424 95-1428 95-1431 *95-1434 *95-1442 105-1842 105-1846
TPS	= 177564			#14-362 26-562 27-568 28-574 29-582 29-589 30-596 30-607
TRAPA	= 000010			96-1447 #96-1452
TRAP10	020476			#14-356 *92-1380 92-1384
TRCSR	= 177560			*99-1681 *99-1683 99-1691 99-1707 #100-1733
TRP240	022432			96-1445 #96-1450
TRP244	020466			#14-349 48-818 49-824 50-830 51-838 51-846 52-853 52-864 75-1110
TRT	= 000003			96-1449 #96-1460 96-1485
TSFCIS	020520			20-418 #22-432
TST1	000652			28-571 28-576 #29-577
TST10	002620			

SYMBOL	CROSS REFERENCE	REFERENCES		
SYMBOL	VALUE			
TST100	015422	84-1187	84-1198	#85-1199
TST101	015744	85-1199	85-1225	#86-1226
TST102	016114	#87-1240		
TST103	016240	#88-1254		
TST104	016734	#89-1324		
TST105	017112	#90-1340		
TST106	017300	#91-1357		
TST107	017466	#92-1376		
TST11	002776	29-577	29-591	#30-592
TST110	017622	92-1376	92-1385	#93-1387
TST111	017746	#94-1400		
TST112	020152	#95-1421		
TST113	020374	#96-1444		
TST114	021140	#97-1558		
TST115	021414	#98-1587		
TST116	021750	#99-1652		
TST117	022630	99-1706	99-1717	#101-1778
TST12	003334	#31-618		
TST13	003440	31-618	#32-626	
TST14	003540	32-626	32-631	#33-632
TST15	003642	33-632	33-637	#34-638
TST16	004016	34-638	34-651	#35-652
TST17	004342	35-652	35-676	#36-677
TST2	001232	22-432	22-481	#23-482
TST20	004470	#37-688		
TST21	004562	37-688	#38-694	
TST22	004662	38-694	38-699	#39-700
TST23	004764	39-700	39-705	#40-706
TST24	005142	40-706	40-720	#41-721
TST25	005500	#42-747		
TST26	005572	42-747	#43-753	
TST27	005672	43-753	43-758	#44-759
TST3	001612	23-482	23-521	#24-522
TST30	005774	44-759	44-764	#45-765
TST31	006152	45-765	45-779	#46-780
TST32	006472	46-780	46-802	#47-803
TST33	006620	#48-815		
TST34	006712	48-815	#49-821	
TST35	007012	49-821	49-826	#50-827
TST36	007114	50-827	50-832	#51-833
TST37	007272	51-833	51-848	#52-849
TST4	002102	24-522	24-544	#25-545
TST40	007630	#53-882		
TST41	007722	53-882	#54-888	
TST42	010022	54-888	54-893	#55-894
TST43	010124	55-894	55-899	#56-900
TST44	010302	56-900	56-914	#57-915
TST45	010622	57-915	57-937	#58-938
TST46	010714	58-938	#59-944	
TST47	011014	59-944	59-949	#60-950
TST5	002324	25-545	25-558	#26-559
TST50	011116	60-950	60-955	#61-957

SYMBOL	CROSS REFERENCE	REFERENCES							
SYMBOL	VALUE								
TST51	011274	61-957	61-971	#62-972					
TST52	011612	62-972	62-994	#63-996					
TST53	011774	#64-1011							
TST54	012076	64-1011	64-1016	#65-1017					
TST55	012202	65-1017	65-1022	#66-1023					
TST56	012364	66-1023	66-1037	#67-1038					
TST57	012710	67-1038	67-1060	#68-1061					
TST6	002416	#27-565							
TST60	013050	#69-1073							
TST61	013150	69-1073	69-1078	#70-1079					
TST62	013334	70-1079	#71-1106						
TST63	013442	#72-1107							
TST64	013550	#73-1108							
TST65	013656	#74-1109							
TST66	013764	#75-1110							
TST67	014072	#76-1111							
TST7	002516	27-565	27-570	#28-571					
TST70	014220	76-1113	#77-1114						
TST71	014326	#78-1116							
TST72	014464	#79-1132							
TST73	014570	79-1132	#80-1141						
TST74	014714	80-1141	80-1150	#81-1151					
TST75	015026	81-1151	81-1158	#82-1160					
TST76	015136	82-1160	82-1168	#83-1170					
TST77	015276	83-1170	#84-1187						
TTCSR	= 177564	#14-355	*89-1332	*89-1336	*90-1346	*90-1355	*91-1362	*92-1379	92-1382 *94-1408
		*94-1416	*94-1419						
T24TRP	022412	99-1658	#99-1724						
T4TRP	022376	99-1656	#99-1721						
ZTRP	022362	99-1654	#99-1718						
\$APTHD	000330	18-391	#18-391						
\$CPUOP	000326	#17-389							
\$DEVCT	000310	#17-389							
\$ENDAD	023042	16-387	20-403	102-1799	#102-1808				
\$ENV	000320	#17-389							
\$ENVM	000321	#17-389	102-1797	104-1831	105-1840				